NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

CHANGING THE PARADIGM: SIMULATION, A METHOD OF FIRST RESORT

by

Ben L. Anderson

September 2011

Thesis Advisor:	Thomas W. Lucas
Second Reader:	Devaushi Singham

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** September 2011	**3. REPORT TYPE AND DATES COVERED** Master's Thesis

4. TITLE AND SUBTITLE Changing the Paradigm: Simulation, a Method of First Resort

5. FUNDING NUMBERS

6. AUTHOR(S) Ben L. Anderson

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES)
N/A

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number N/A.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited

12b. DISTRIBUTION CODE
A

13. ABSTRACT (maximum 200 words)

The computing capability to which Operations Research (OR) analysts have access today is over 1,000,000,000 times more powerful than the first simulation pioneers had sixty years ago, yet the concept that simulation is a "method of last resort" still plagues the OR community. Many real-world problems are complex, with properties such as high dimensionality, non-linear effects, stochastic elements, and dependence between variables. Solving these problems analytically often requires simplifying assumptions, running the risk of making a Type III error (i.e., getting the right answer to the wrong problem). This paper explores the development of computer simulation, and the key design principles that must be followed, to demonstrate how simulation is often the appropriate tool in understanding complex, real-world problems. Contrasting the results of a recently published analytical approach to the analysis of an airport check-in counter scheduling problem versus those of a simulation study of the same system, we demonstrate that simulation can quickly provide the same answers with any desired degree of precision and with no loss of insight. More importantly, simulation can easily use both empirical data and more realistic assumptions—which allows for the analyst to address the right problem. With current computational capabilities and methods, it is time to change the paradigm. Simulation is a method of first resort.

14. SUBJECT TERMS Simulation, Queuing Theory, Design of Experiments (DOE), Object Orienting Modeling, Computational Tractability, Airport Scheduling, Simio

15. NUMBER OF PAGES
105

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

CHANGING THE PARADIGM: SIMULATION,
A METHOD OF FIRST RESORT

Ben L. Anderson
Lieutenant, United States Navy
B.S., North Carolina State University, 2004

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 2011

Author: Ben L. Anderson

Approved by: Thomas W. Lucas
Thesis Advisor

Devaushi Singham
Second Reader

Robert F. Dell
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The computing capability to which Operations Research (OR) analysts have access today is over 1,000,000,000 times more powerful than the first simulation pioneers had sixty years ago, yet the concept that simulation is a "method of last resort" still plagues the OR community. Many real-world problems are complex, with properties such as high dimensionality, non-linear effects, stochastic elements, and dependence between variables. Solving these problems analytically often requires simplifying assumptions, running the risk of making a Type III error (i.e., getting the right answer to the wrong problem). This paper explores the development of computer simulation, and the key design principles that must be followed, to demonstrate how simulation is often the appropriate tool in understanding complex, real-world problems. Contrasting the results of a recently published analytical approach to the analysis of an airport check-in counter scheduling problem versus those of a simulation study of the same system, we demonstrate that simulation can quickly provide the same answers with any desired degree of precision and with no loss of insight. More importantly, simulation can easily use both empirical data and more realistic assumptions—which allows for the analyst to address the right problem. With current computational capabilities and methods, it is time to change the paradigm. Simulation is a method of first resort.

THIS PAGE INTENTIONALLY LEFT BLANK

THESIS DISCLAIMER

The reader is cautioned that the simulation model presented in this thesis has not been exercised and tested to its fullest extent. While every effort has been made within the time available to ensure that the simulation represents the scenario presented by Parlar and Sharafali (2008) (henceforth referred to as the Parlar Model), interpretations of their intentions and distributions were made to facilitate the development of the simulation. Further, this model does not represent a real-world system, and has been made to demonstrate the benefits of computer simulation when examining real-world problems.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS AND ABBREVIATIONS

AMSO	Army Modeling and Simulation Office
AP EXCOM	Adaptive Planning Executive Committee
AT&L	Acquisition, Technology, and Logistics
C4ISR	Command, Control, Communications, Computers Intelligence, Surveillance, and Reconnaissance
CAC	Common Access Card
CAPE	Cost Assessment and Program Evaluation
CAPS	C4ISR Architecture Planning/Analysis System
COCOM	Combatant Commander
COSMOS	C4ISR Space and Missile Operations Simulation
CPU	Central Processing Unit
CSV	Comma-Separated Values
DIMSCG	Defense Intelligence Modeling and Simulation Coordination Group
DoD	Department of Defense
DOE	Design of Experiments
EADSIM	Extended Air Defense Simulation
EOSAN	European Organization for the Safety of Air Navigation
ENIAC	Electronic Numerical Integrator and Computer
FAA	Federal Aviation Administration
FLOPS	Floating Point Operations per Second
GCAM	General Campaign Analysis Model
GEOINT	Geospatial Intelligence
HUMINT	Human Intelligence

ITEM	Integrated Theater Engagement Model
JCDE EC	Joint Concept Development and Experimentation Executive Committee
JFCOM	Joint Forces Command
JIMM	Joint Integrated Mission Model
JS	Joint Staff
JTLS	Joint Theater Level Simulation
M&S	Modeling and Simulation
M&S SC	Modeling and Simulation Steering Committee
MSCO	Modeling and Simulation Coordination Office
NMSO	Navy Modeling and Simulation Office
NSS	Naval Simulation System
OR	Operations Research
OSD	Office of the Secretary of Defense
OUSD	Office of the Under Secretary of Defense
P&R	Personnel and Readiness
ROI	Return on Investment
SE	Systems Engineering
SIGINT	Signals Intelligence
SSA SC	Standard Simulation Architecture Steering Committee
Simio	Simulation Modeling framework based on Intelligent Objects
STORM	Synthetic Theater Operations Research Model
SPL	Simulation Programming Language
T2 ESG	Training / Transformation Executive Steering Group
T&E BOD	Test and Evaluation Board of Directors
TACWAR	Tactical Warfare Integrated Environment

USAF	United States Air Force
USD(I)	Under Secretary of Defense (Intelligence)
V&V	Verification and Validation
VIC	Vector-in Commander
VV&A	Verification, Validation, and Accreditation
WSC	Winter Simulation Conference

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

> If anyone, here or later, can tell us how the approach of certainty—traditional mathematics—is going to answer the questions that practical data analysts are going to have to have answered, I will rejoice. But until I am reliably informed of such a utopian prospect, I shall expect the critical practical answers of the next decade or so to come from the approach of simulation—from a statistician's form of mathematics, in which ever more powerful computing systems will be an essential partner and effective. . .
>
> —John Tukey, (1986)

The paradigm that simulation is a method of last resort has been around for over 50 years, with its first use in operations research focused literature by John Harling (1958), in an article entitled "Simulation Techniques in Operations Research." In that article, Harling uses the term "it has been often said," implying that the paradigm of simulation being a tool of last resort was not just prevalent at the time, but had been for several years (Harling, 1958). The phrase saw continued use throughout the operations research community, and is perhaps best known from Harvey Wagner's seminal textbook *Principles of Operations Research* (Wagner, 1969). Why was this the paradigm of the time? Was it a valid paradigm at the time, and is it a valid paradigm now?

Since the inception of computer simulation with the use of the Electronic Numerical Integrator and Computer (ENIAC), there have been significant advances in both processing power and storage capability. Computing power today is over 1,000,000,000 times what it was just 60 years ago. It took the ENIAC's computational capabilities to solve the complex integrals needed in the design of a thermonuclear weapon, and with the increases in computing power, many more computationally challenging problems have now become computationally tractable. We define "computationally tractable" as follows: A model is computationally tractable if it can be solved on a computer to any required level of precision in a short period of time.

Unfortunately, some in the OR community have not fully embraced the power of today's computers, and has failed to recognize the benefits of simulation to its fullest. One may argue that simulation can only provide an approximate solution, while an

analytical one presents an exact one. That statement, however, needs to be placed into its proper context. An analytical solution may only be exact when it is done in the absence of assumptions. The moment one must "assume," is the moment the analytical solution slips away from the paramount of perfection, to simply providing an "approximate" solution.

This thesis provides an examination of a paper written by Parlar and Sharafali (2008) on an analytical approach to providing an optimal solution on the allocation of airline check-in counters. Numerous assumptions were made by Parlar and Sharafali in order to present an analytically tractable problem. Many of which are made contrary to the actual workings of the system. Three of the most significant assumptions are:

- the time period in which the counters are open is a fixed time (T) that can be partitioned into (K) segments, in which the arrival rate can be estimated as a constant;
- service times are exponentially distributed; and
- a linear increase in service rate, based on the number of passengers in the system who have not completed service.

These assumptions have a dramatic impact on the authors' objective function to be minimized. This objective function, which is evaluated through the use of a dynamic program, calculates the minimum expected cost to go from time t_k to T, given m arrivals and, of those arrivals, n having completed service. This expected cost function is significantly affected by the distribution of service times and the linear increase in service rate.

With an understanding of the model presented by Parlar and Sharafali, this thesis then presents the methodology presented originally by Law and Kelton (1982), and adopted by the Defense Modeling and Simulation Coordination Office for Verification, Validation and Accreditation (Modeling and Simulation Coordination Office (MSCO), 2001). Through this process, and the use of Simio simulation software, this thesis produced a simulation model, copying the assumptions and distributions of the Parlar and Sharafali model as closely as possible. As this model is computationally tractable, through simulation we quickly were able to replicate the transition probabilities

calculated by Parlar and Sharafali to any degree of precision required. The simulation model was then exercised to test the validity of Parlar and Sharafali's findings of the optimal configurations, the monotonicty within the optimal solution, and that the number of counters to open is nonincreasing in the number of passengers who have been serviced.

Contrary to Parlar and Sharafali's findings, our results demonstrated that not only did monotonicity not exist in the solution, but the "optimal" solution calculated by the dynamic program did not have the lowest mean cost. To investigate this, we examined two of the key the assumptions made by Parlar and Sharafali. First, the linear increase in service rate was evaluated. We demonstrated that this assumption played a significant role in the development of the optimal solution and, through a numerical illustration, is not a reasonable assumption. The second demonstration illustrated the importance of choosing the right distribution model. Utilizing a more realistic distribution, with the same mean value, demonstrated that the average total cost of a run is lower, and from a much narrower range, than when calculated with the exponential distribution due to the heavier weightings of the exponential distribution at higher numbers.

With today's computing power and storage capacity, combined with a logical and formal method for developing and validating simulation studies, simulation should no longer be looked at as a method of "last resort." No longer are analysts constrained to having to make assumptions such as normality, independence, memorylessness, deterministic, linear, stationary, and homoscedasticity—as the advanced simulation suites handle these conditions with ease. No longer do analysts have to manipulate a problem to make it analytically tractable, which can lead the analyst away from solving the real problem as they are preoccupied by solving the analytical one. Simulation allows for a more realistic analysis to be done, being able to incorporate actual statistics of the real systems into the simulation. It allows for the simultaneous exploration of multiple parameters of the system through a proper design of experiments. It provides the insight needed by decision makers to make robust decisions. It is time to change the paradigm. Simulation is a method of first resort.

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

The opportunity to attend the Naval Postgraduate School (NPS) has been an incredible step in my life's journey. Eighteen years ago (September 1993), I departed my native land of Canada and arrived upon the hallowed yellow footprints at Marine Corps Recruit Depot San Diego. From my first day of Boot Camp, to the graduation ceremonies here at NPS, the military has never stopped providing me with opportunities to learn and develop as an American citizen, as a member of the armed forces, as a leader, as a husband, and as a father.

Throughout my eighteen years of service, it is neither what I have personally achieved, nor the places that I have been which have guided me to who I am today. It has been the people whom I have met, worked for, and worked with that have made this journey a wonderful, enjoyable, and enriching experience.

The two people who have provided the most significant impact on my life are my wife and daughter. My wife, Laurel, has provided me with a stability I never would have thought possible. Taking care of the all the details of life, while enduring my many absences due to military obligations, with nothing but a "can-do" attitude, has enabled me to work harder and focus more on the tasks at hand, rather than having to worry if all was being taken care of on the home front. My daughter London motivates me to ensure that the decisions I make will ensure the long-term success of our family. Without the two of them standing with me and supporting me, I doubt I would have been able to achieve what I have.

Several key military leaders have had a more profound impact on my development than others. First, Corporal (USMC) Joel Montejano taught me that no matter how hard one worked, unless one could work well with others—the benefit of the hard work would never be realized. Second, General (USA, Retired) Henry Hugh Shelton, through his seminar on "Transformational Leadership" at the Shelton Leadership Conference at North Carolina State University, taught me that while the "status quo" may be what is comfortable, a true leader has to seek improvements, even when they are not

desired. Finally, Captain (USN) David Honabach, my commanding officer while serving on the USS Jimmy Carter (SSN-23), has two sayings which I failed to fully appreciate at the time, but now hold in high regard:

> *“Communication is the key to success, lack of communication is the key to failure.”*
>
> *“Some training is good, more training is better, too much training is just right.”*

Dr. Thomas Lucas, my advisor and mentor, has been instrumental in my development as a scholar. Through stimulating conversations and enriching opportunities to concise and constructive criticism, he has challenged me to broaden my viewpoint, bolster my arguments, and realize when I have sufficiently proven my point. The lessons learned through his class, and the opportunity he provided via International Data Farming Workshops, has greatly enriched my life.

To accomplish this thesis, two others have contributed significantly in formulating the thought process, reasoning the arguments, and vetting my many grammatical errors. I give my sincere appreciation to both Dr. W. David Kelton and Dr. Dashi Singham, for their help, mentoring, and critiquing throughout this experience.

At NPS, several other professors taught me much more than what was indicated on their respective syllabi. Dr. Paul Sanchez, Dr. Casey Lucius, and Captain (USN, Retired) Jeffrey Kline have all taught me (albeit in different subjects), that sometimes the most important facts are the ones that aren’t mentioned and that true understanding can only come from delving into the unknown, the unexplained, the unmentioned, or the unreasoned.

“Ex umbris et imaginibus in veritatem” Plato

(*Out of shadows and semblances into the truth*)

I. INTRODUCTION

It has been often said that a simulation is a last resort.

—Harling, (1958)

A. BACKGROUND AND LITERATURE REVIEW

The paradigm that simulation is a method of last resort has been around for over 50 years. The earliest record of the phrase (as quoted above) is from 1958, when John Harling gave a presentation to the London School of Economics on operations research. One must note that Harling uses the term "it has been often said," thus implying that the paradigm of simulation as a tool of last resort was not just prevalent at the time, but had been for several years (Harling, 1958). The phrase saw continued use throughout the operations research community, and is perhaps best known from pages 887 and 890 of Harvey Wagner's seminal textbook *Principles of Operations Research* (Wagner, 1969). Why was this the paradigm of the time? Was it a valid paradigm at the time, and is it a valid paradigm now?

First, to discuss the term simulation, one must be clear of the definition being used. In the case of this paper, simulation strictly refers to the use of a numerical model to study the behavior of a system as it operates over time using a computer (Kiviat, 1967). To have a clearer understanding of the term in the operations research context, one should turn to the explanation provided by John E. Cremeans in his 1967 paper entitled "Why Simulation?" Cremeans, rather than defining simulation in a concrete way, chose to provide a characterization of what a simulation is and is not. First, Cremeans clarifies that simulations are not optimizing programs, and that only through data analysis of the simulation output can one gain insight into the model and apply that insight in an effort to optimize the system. Second, Cremeans compares simulation to that of a laboratory experiment, where the variables can all be controlled and altered as seen fit by the programmer. In fact, a stochastic simulation can be run multiple times with all the parameters remaining the same, or changing them as the designer sees fit, which is different from virtually all true laboratory experiments.

From the Buffon needle experiments in the 1770s to estimate π, to the complex models run on supercomputers and computing clusters today, simulation persists and has grown in its use, but the phrase "as a last resort" can still be seen in scholarly papers. One should examine how simulation has evolved over time in order to make the determination if the adage is still valid.

1. A Brief History of Computer Simulation

> If you would understand anything, observe its beginning and its development.
>
> —Aristotle

Keeping with the academic spirit of Aristotle, let us examine the development of simulation over time in order to gain an appreciation of the history of simulation and how it has changed with the advances of technology. To examine the history of simulation, Goldsman, Nance, and Wilson (2009) have provided a framework with which one can segregate the changes of computing power over time and its effect on simulation. Their framework consists of three definitive phases: the Precomputer Era (Pre–1945), the Formative Period (1945–1970), and the Expansion Period (1970–1981). Their study of simulation history stopped at 1981, as they did not believe that sufficient time had passed to see any definitive changes since that time (James R. Wilson, personal communication, August 11, 2011). However, during a discussion with Professor James Wilson of North Carolina State University regarding the substantial increases in computing power and memory availability, it was proposed that two additional periods may be examined: the Maturation Period (1983–2000 [approximately]) and the Distributed Processing Era (2000–present).

a. Precomputer Era

> The idea behind the Monte Carlo approach. . . is to [replace] theory by experiment whenever the former falters.
>
> —Hammersley & Handscomb, (1964)

Although the term "Monte Carlo Simulation" was not known or discussed at the time, the process we know today as "Monte Carlo Simulation" has been utilized as

far back as 1777, when Buffon conducted his famous "needle experiment" to estimate the value of π (Goldsman, Nance, & Wilson, 2009). Further, in 1908, William Sealy Gosset used a manual simulation to verify the probability density function for his "Student's 't' Distribution" prior to his seminal paper's 1908 publication (see Figure 1). It is important to note that these "simulations" conducted by Buffon and Gosset were both validated later through analysis; Buffon's by Laplace (Laplace, 1812) and Gosset's by Fisher (Fisher, 1925).

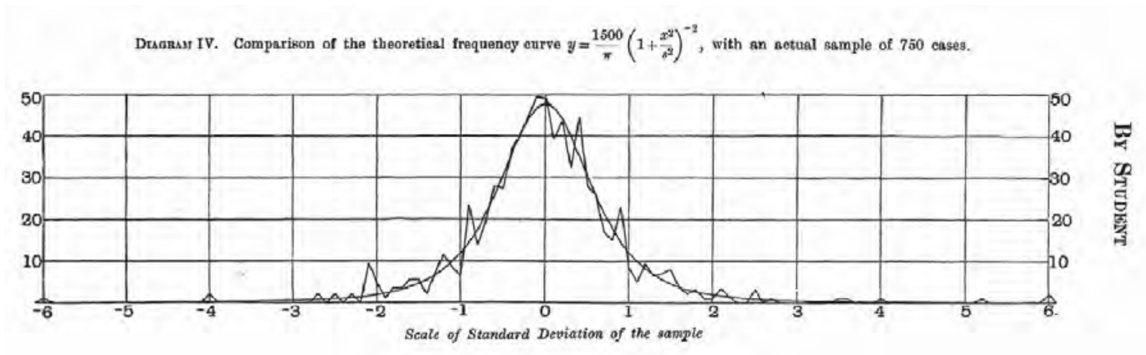


Figure 1. Student's Comparison of Theoretical Frequency Curve With an Actual Sample Originally Published in Biometrika (From Student, 1908)

b. *Formative Period*

It was John Von Neumann, Stanislaw Ulam, and Nicholas Metropolis who can take credit for coining the phrase "Monte Carlo methods" when the trio were working on the issue of neutron diffusion and also realized the computational potential of the computers being built (Metropolis, 1987). Recognizing the challenge in the analytical analysis of the diffusion of particles and either particle procreation or multiplication, the trio created a mathematical description of the problem that consisted of both deterministic and stochastic processes. Then, through the use of random sampling, they were able to obtain sample sets of the results for a statistical analysis. This "Monte Carlo method" was adopted by many more people as expanded access to computers became a reality.

What enabled the trio to develop their "Monte Carlo method" was the invention of America's first computer—the ENIAC (Electronic Numerical Integrator and Computer). Using their method and formulas, the trio submitted their calculations for

computation by ENIAC and, with the results, were able to design a working thermonuclear weapon. It is essential to recognize that prior to the ENIAC, the complex integrals needed to design the weapon where analytically intractable (Fritz, 1994). These calculations became computationally tractable only through the building of the ENIAC and the development of the "Monte Carlo method" to solve them.

With the potential of computer simulation just starting to be realized, the period from 1945 through 1970 saw the development of computing capacity all over the world in government, corporate, and academic institutions. Not only was the access to computing facilities becoming readily available to researchers and scientists, others began to develop programming languages to better facilitate the use of these machines. For example, FORTRAN was developed between 1954 and 1957 in order to automatically generate efficient machine code from an easier to read input design (Padua, 2000).

As the use and availability of simulation grew, so did the understanding of the difficulties associated with simulation. In order to confront these difficulties, Conway, Johnson, and Maxwell of Cornell University addressed them in two key papers in 1959 and 1963. They identified two components to the use of computer simulation that needed to be examined: the construction of the model and the analysis of the results. These papers identified issues in both components, some of which have been resolved, while others are still being researched.

c. Expansion Period

Up until the late 1970s, computer access time was costly and severely limited to institutions that could afford the large computers. It was the advent of the "personal computer" that brought the computing power needed for computer simulation to the average user.

As the availability of computers grew, so did the number of available simulation programming languages (SPLs). The primary source of improvement through the 1970s was the development of more efficient and user friendly SPLs. As the number of SPLs increased, the textbooks also began to change. Several key texts were published

that were not arbitrarily tied to a specific SPL and focused on computer simulation as a method in general (e.g., Law & Kelton, 1982).

In addition to the improvement of SPLs, several other key topics were advanced during this period. A new object-oriented approach to simulation design entitled "Conical Methodology" was developed by Nance (1978). Additionally, advances in random number generation, event graphs (Schruben, 1983), and the need for formal verification and validation (Blaci & Sargent, 1981) were brought forward. The text *Simulation Modeling and Analysis* by Law and Kelton (1982) has been credited with being the first of many that brought the advanced methodologies to a wide audience (Goldsman, Nance, & Wilson, 2010).

d. Maturation Period

With a definitive structure set in place on the issues that face the use of simulation, combined with an ever increasing source of experts, research continued throughout the maturation period on how one can achieve more reality and fidelity in simulation. One source which has had a significant influence, and continues to underpin these developments, is the Winter Simulation Conference (WSC).

While the WSC provides a forum for discussion of the most recent advances in simulation, it also "provides the central meeting place for simulation practitioners, researchers, and vendors working in all disciplines and in the industrial, governmental, military, service, and academic sectors" (White, Fu, & Sanchez, 2011). The WSC has annually brought together individuals and sponsors from six major professional organizations and one government agency to accomplish this through a completely volunteer lead effort for over 40 years (White, Fu, & Sanchez, 2011). For more information on the WSC, and access to their past presentations and papers, refer to http://wintersim.org/.

The Department of Defense's (DoD) reliance on modeling and simulation grew throughout the 1990s. It was during this period that the use of simulations for operational wargaming became widespread (Army Modeling & Simulation Office [AMSO], 2011). With that usage came the realization that without proper quality control

being applied, the validity of the models and simulations would be dubious. To address this, and several other issues, the DoD published Directive 5000.59 in 1994, mandating all services and defense agencies to establish a modeling and simulation (M&S) office in order to provide a structure in which future M&S initiatives would be developed and controlled.

e. Distributed Processing Era

Goldsman, Nance, and Wilson (2009) stopped their detailed analysis of the history of simulation at the end of the expansion era, as it was felt that "insufficient history had been accumulated" since that time to effectively comment on the maturation period (James R. Wilson, personal communication, August 11, 2011). However, during a phone interview with Professor Wilson, it was suggested that the maturation period was over and that the Distributed Processing Era had begun. Professor Wilson not only agreed that this was the case, but stated that entering into the Distributed Processing Era may be as significant to the simulation community as the realization of the potential of the ENIAC back in 1946 (James R. Wilson, personal communication, August 11, 2011).

The ability to run simulations on multiple processors simultaneously is not a new concept or capability; however, the ability to do so on a relatively inexpensive machine is. With multicore architecture now the standard on processors, simulation software companies who specialize in the development of software for those who do not have access to computer clusters are beginning to exploit the capabilities of the new multicore architectures. For example, a simulation software suite entitled Simio (Simulation Modeling Framework based on Intelligent Objects), having recognized the greater performance that can be achieved through using the multicore architecture, have included the following statement in their hardware requirements page: "Multiple scenarios and replications run in the Experiment Window take full advantage of each available processor (e.g., a quad-core processor will execute 4 replications or scenarios simultaneously" (Simio, 2011).

2. Changes in Computer Architecture and Cost

> The new slope might approximate a doubling every two years, rather than every year, by the end of the decade.
>
> —Gordon Moore, (1975)

a. Application of Moore's Law

While many people quote Moore's "law," there are many who quote him incorrectly. Mr. Moore provided clarification of what Moore's "law" entails in a video interview he did with Intel in 2005. Moore clarified that while his original prediction of transistor density was correct at the time (i.e., the number of transistors per chip were doubling twice per year), that it was only valid until 1975. In 1975, he reevaluated his estimate to complexity doubling once every two years. He also asserted that not only has it proven to be true, but it is now a driver behind the progress, as industry holds it as a standard to live up to. Figure 2 shows a plot of transistor count to production year, from 1971 through 2008.

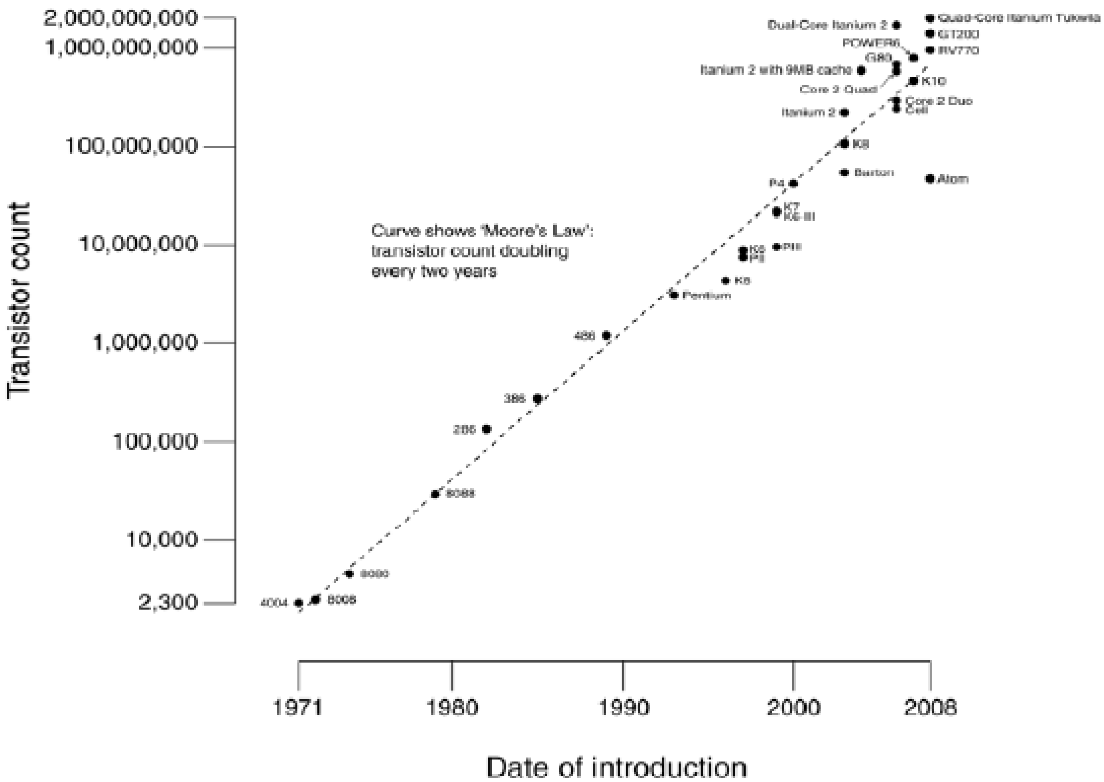


Figure 2. Moore's Law Depicted Through the Change in Transistor Count Over Time (From Moore's Law, 2011)

This is of significance to the operations research community, as the ability to compute complex numbers and run complex simulations is highly dependent on the processing capability of the computer system being used. In a study conducted by the Lawrence Berkeley National Laboratory in 2008, the rate of increase in floating point operations per second (FLOPS) has also been consistent with Moore's "law" (see Figure 3). We now have over one billion times more processing power than those first simulation pioneers had 60 years ago.

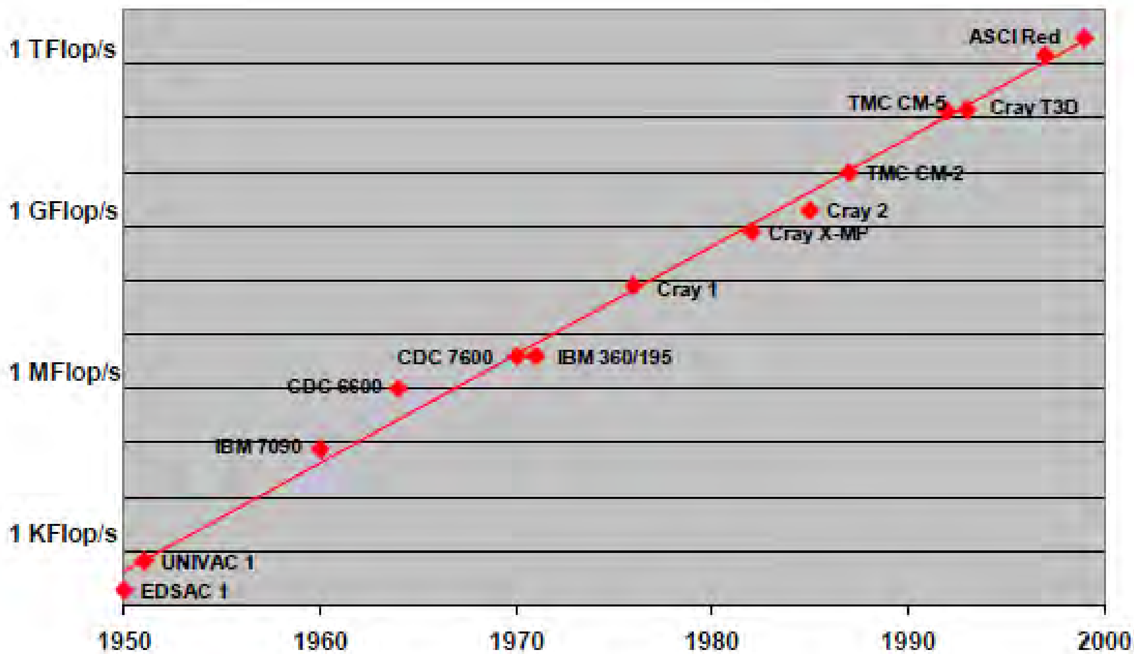


Figure 3. Moore's Law and Peak Performance Over Time (From Dongarra, Meuer, Simon, & Strohmaier, 2000)

b. Relative Cost and Quantity of Memory

In addition to the increase in the computing performance factor of FLOPS, there has been an equally dramatic change in the availability and cost of data storage over time. This decrease in cost has allowed for much more data to be retained than before, and at a much lower cost. Using historical data, Mr. Matthew Komorowski, a software engineer from Buffalo, New York, found that the amount of storage per unit cost has shown to double approximately every 14 months (Komorowski, 2009).

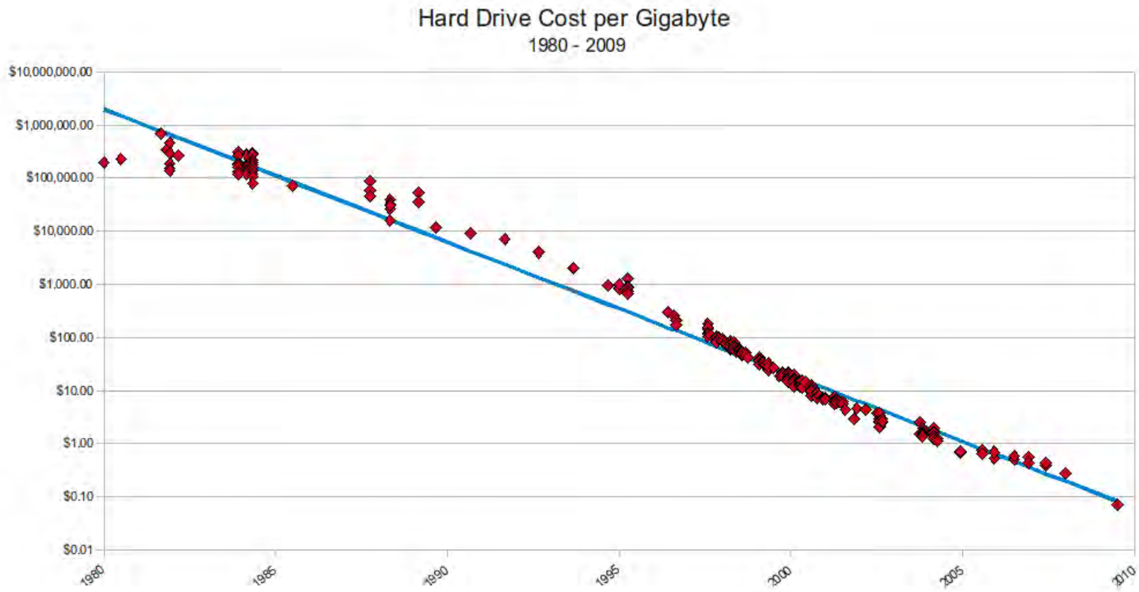


Figure 4. Hard Drive Cost per Gigabyte (From Komorowski, 2011)

c. *Computational Tractability*

> Analytical procedures are usually preferred over numerical ones, as they are more accurate and less costly to compute.
>
> —Kiviat, (1967)

With the increase in processing power and available memory space, problems that once were too complex to be solved are beginning to become computationally tractable. Through this increase, the ability to use analytical procedures has also increased, as well as the ability to run simulations with much higher degrees of reality and fidelity than previously seen.

For example, Parlar and Sharafali show the equation used to calculate the transient probability that for a given set of conditions, the probability to move from state (m,n) to a state (i,j) can be determined by the following equation (for additional details, refer to Parlar and Sharafali, 2008):

$$P^{c}_{m,n}(i,j,t) = \binom{N-m}{i-m} e^{-(N-i)\lambda t} \cdot \sum_{r=0}^{\max(i-m,m-n)} \binom{i-m}{r} \left[\alpha(c,t)\right]^{i-m-r} \left[\beta(c,t)\right]^{r} \times \binom{m-n}{j-m-r} e^{-(m-j+r)c\mu t} \left(1-e^{-c\mu t}\right)^{j-n-r} .$$

With the conditions of $N = 10$, $c = 1,2,3$ with $(m,n) = (4,2)$, $(i,j) = (7,3)$, and $(\lambda,\mu) = (1.5,5)$, they produced the following plot of transient probability versus time (see Figure 5).

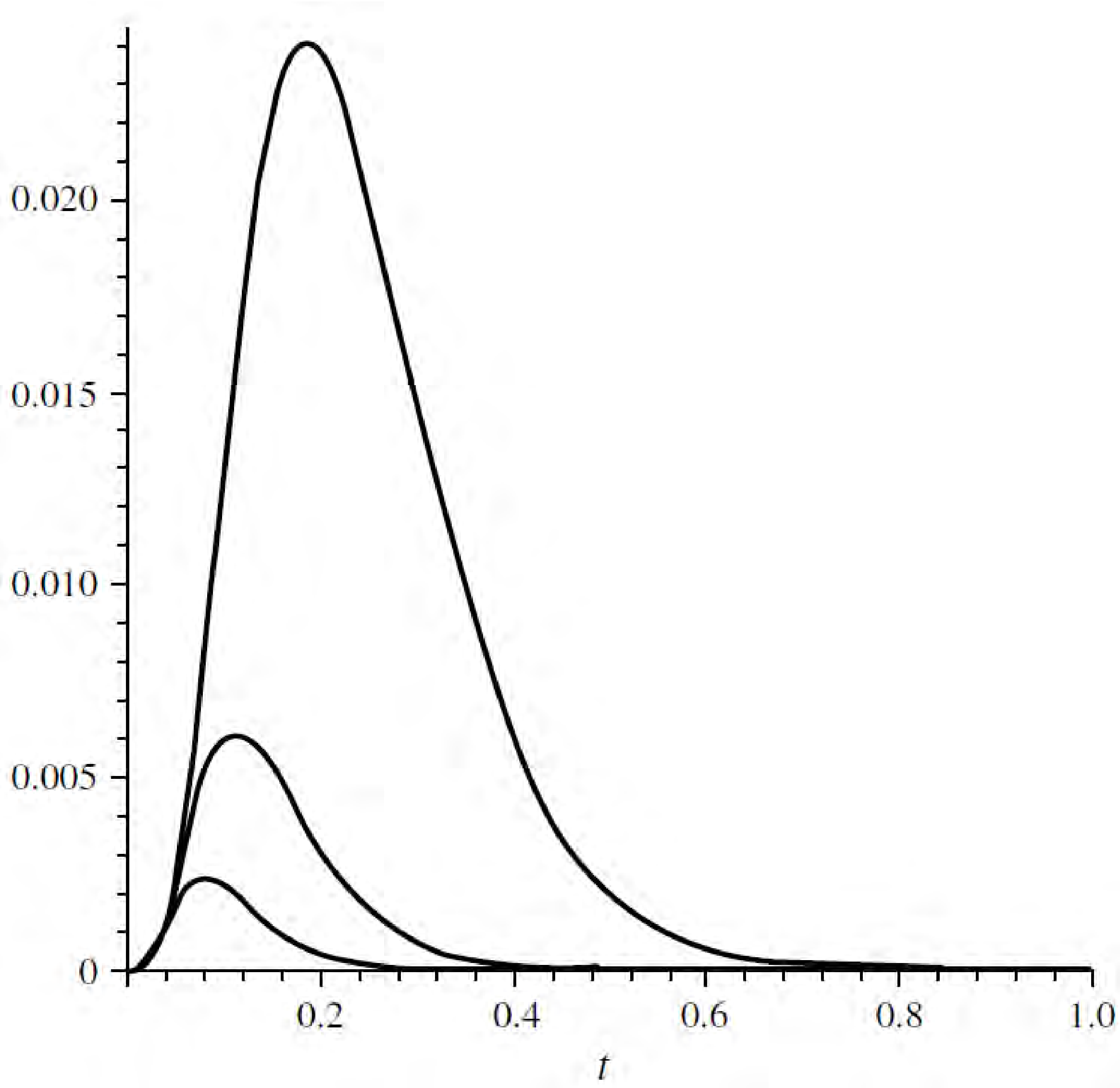


Figure 5. Transient Probability Graph With $c = 1$ Having the Highest Peak (From Parlar & Sharafali, 2008)

This same plot was found through a simulation approach implemented in MATLAB (http://www.mathworks.com/products/matlab/). The simulation was

developed through an examination of the transition rate diagram (see Figure 6) provided by Parlar and Sharafali (2008), which was used to develop their conceptual model.

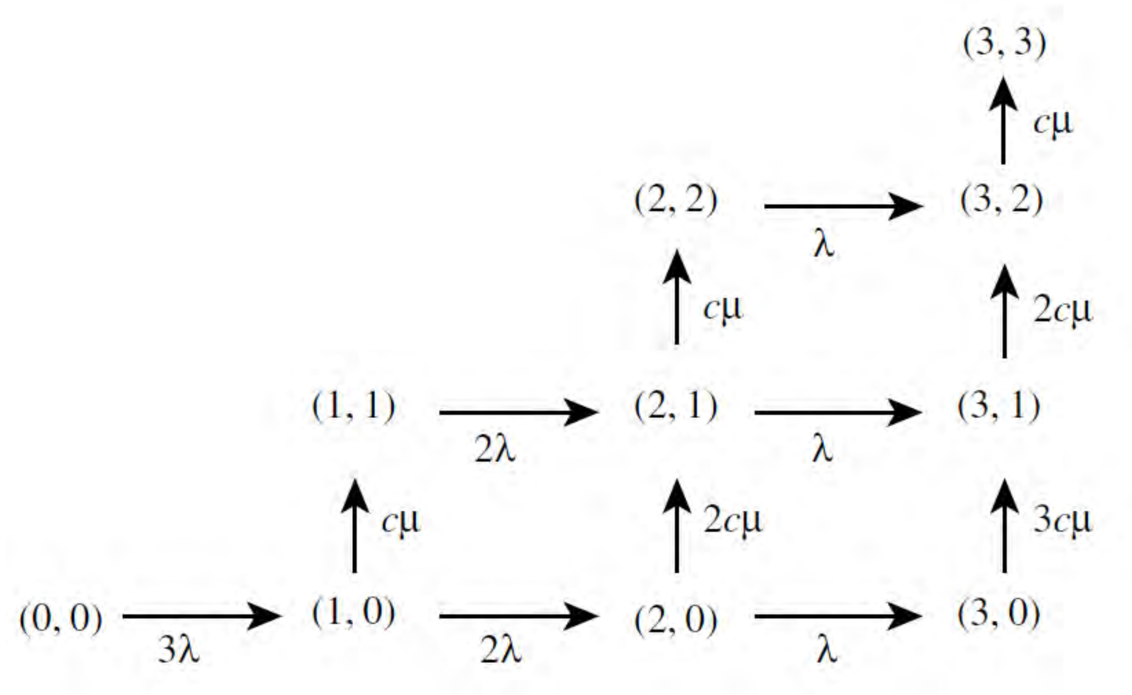


Figure 6. Transition Rate Diagram for $N = 3$ (From Parlar & Sharafali, 2008)

In approximately 30 minutes, (due mainly to a lack of familiarity of the MATLAB syntax), the transition rate diagram shown above was implemented in MATLAB and an initial 1,000 runs were made to verify that the MATLAB implementation was producing results consistent with Figure 5. These initial runs took approximately 0.1 seconds to run. When it was found that the results matched the expectations, the simulation was run 10^7 times (which took approximately 18 minutes on a desktop computer utilizing an Intel Core i5-2500K central processing unit (CPU), with 8.0 gigabytes of installed memory) and the graph in Figure 7 was produced.

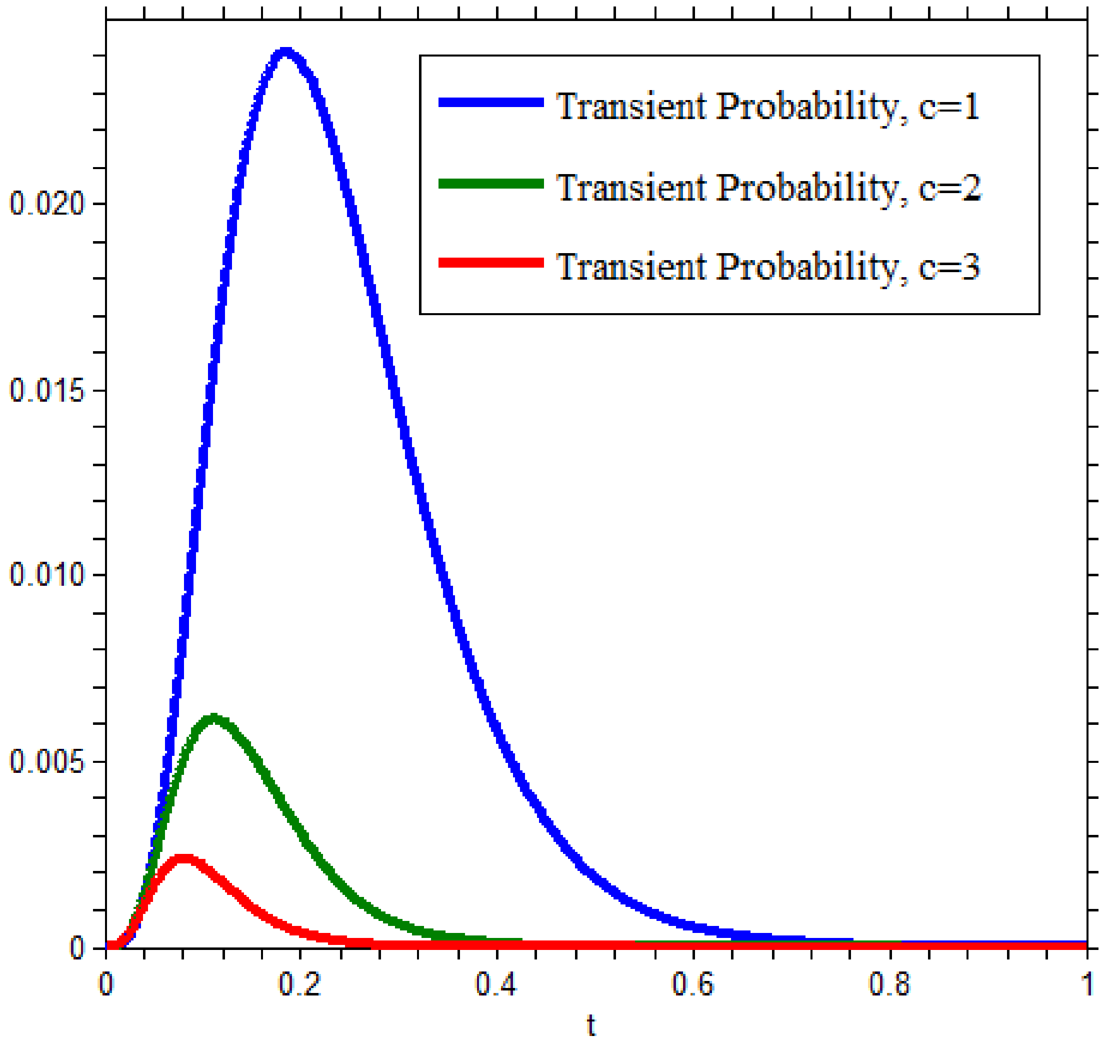


Figure 7. Simulation Result for Transition Probability Graph

The displayed graph produced by the MATLAB simulation appears to be identical to the results produced by the analytical, yet the simulation does not provide an "exact" answer. However, when the confidence interval for the simulation results is smaller than the width of the line depicting the distribution, how relevant is it? In this case, the 99% confidence interval half-width is bounded as follows (with p determined as the maximum value found at 0.0241 and n being the number of trials set at 10^7):

$$CI_{99\%} \text{ half-width} = \Phi^{-1}(0.995)\sqrt{\frac{p(1-p)}{n}} = 0.000125 .$$

With the capability to quickly run the simulation a vast number of times in a relatively short time span, the simulation results converge on the analytical results with such closeness that one must determine when it is “close enough” to count as an equivalent answer. It is also important to note that while asserting the superiority of analytical solutions due to them being more precise, the calculations are most often being made by computer, which (unless conducting strictly integer operations or symbolic manipulation) contain a degree of error to the manner in which real arithmetic operations are conducted.

3. Use of Simulation in the Department of Defense

The DoD has been on the forefront of computer simulation use since its inception with the ENIAC and the Monte Carlo method being used to solve the complex integrals required for the design of the thermonuclear bomb. A recent study conducted by the National Research Council, and published in a book entitled *Defense Modeling*, *Simulation*, *and Analysis: Meeting the Challenge*, recognizes that through the increase in computing power, richer approaches in modeling and simulation have been made possible (Committee on Modeling and Simulation for Defense Transformation, 2006). They illustrate this through examples of simulation areas at three levels of complexity (see Table 1).

Aspect	**Simplistic**	**Intermediate**	**Advanced**
Number of parties	Two sides, with allies folded into the appropriate side	Plus some explicit modeling of third countries	Plus nongovernment organizations and threats
Nature of Variables	Only objective variables, such as a side’s firepower	Plus soft variables such as a side’s fighting effectiveness, affected by leadership and other factors	Plus soft variables such as nationalism, ethnic group association, and propensity for brutality and terrorism

Table 1. Partial List of Levels of Combat Model Sophistication (From Committee on Modeling and Simulation for Defense Transformation, 2006)

The use of simulation has grown to span virtually every component of the DoD. The projected spending for DoD training, training support, and simulation is estimated to total nearly $35 billion annually (MSCO, 2010). In 2010, the DoD M&S workforce consisted of approximately 30,000 military personnel, government civilian employees, and contractors (MSCO, 2010). In addition to the DoD employees, each branch of service has created their own M&S office and separate human capital strategy plan for M&S (see www.ms.army.mil/, nmso.navy.mil/, www.mccdc.usmc.mil/MCMSMO/, and www.afams.af.mil/). Additionally, each service has established a virtual catalog of models and simulations applicable and employed by their respective service. For example, within the Navy's Model and Simulation catalog (accessible via https://nmso.navy.mil/NavyMSRR/Browse/tabid/85/Default.aspx), there are currently 924 simulation models being utilized. Figure 8, which represents the DoD's M&S Governance, shows how vested the DoD is in simulation.

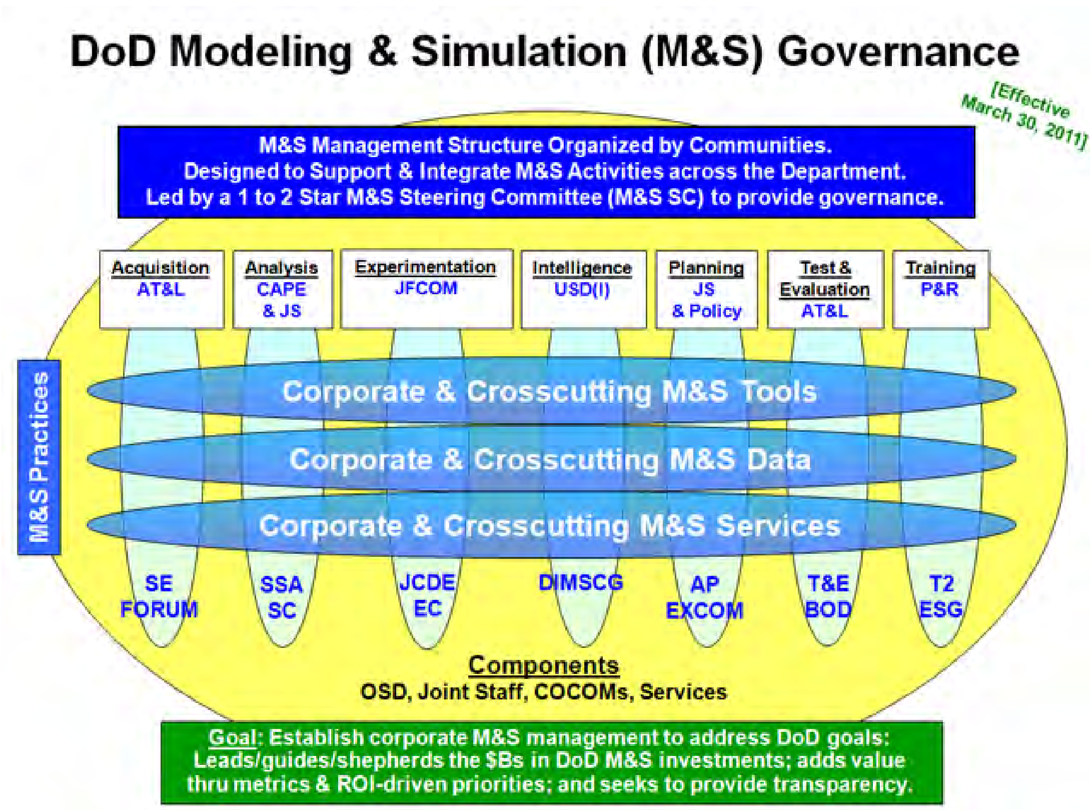


Figure 8. DoD M&S Governance (From MSCO, 2011)

a. Acquisition and Logistics

M&S provides virtual and constructive test beds through which weapon, equipment, and ammunition factors can be prototyped, tested, and evaluated during the acquisition process. M&S develops a level of understanding of the interaction of the parts of the logistical system, and of the logistical system as a whole, which is seldom achievable via any other process. The use of M&S reduces testing time and costs, and allows measurement of phenomenon that cannot be measured using traditional methods. The results of these tests provide data on which procurement decisions are based. Additionally, it allows for the selection and characterization of optimal material solutions. On the logistics side, things such as deployment timelines and studies on how to improve the efficiencies of logistical operations are conducted. The Marine Corps Acquisition M&S community provides access to anyone with a Common Access Card (CAC) with DoD certificates to see their M&S services, tools, and data via their website (linked through https://www.mccdc.usmc.mil/MCMSMO/acquisition.htm).

b. Analysis

The analysis community employs M&S to analyze the performance, effectiveness, survivability, trade-offs, and cost/benefit on everything from logistical and personnel systems to force structure and risk. Additionally, tests are done to determine the effectiveness of weapon systems, performance and system characteristics of equipment and system purchases, and the effectiveness of force management decisions. The DoD's integrated support activity, the Modeling and Simulation Analysis Center, can be reached at http://www.dod-msiac.org/).

c. Experimentation

Through the exploration, testing, and validation of warfighting ideas, insight is found into how we will transform our forces and potentially fight future battles. This is done through experiments involving soldiers and leaders within live, virtual, and constructive environments (see http://www.nps.edu/research/TRAC/).

d. Intelligence

M&S helps provide actionable events designed to stimulate/simulate the proper human intelligence (HUMINT), signals intelligence (SIGINT), and geospatial intelligence (GEOINT), which was gathered in real-world operations. M&S is also used in training intelligence personnel for counterinsurgency operations.

e. Testing

With changes in the DoD instruction for acquisition, M&S is required to be used throughout a program's life cycle (Office of the Under Secretary of Defense (Acquisition, Technologies, and Logistics [OUSD-AT&L], 2006). This is done in order to support requirements definition, the design and engineering phase, test planning, rehearsal, and then the conduct of actual tests. It also is used to aid in evaluating the performance of tested items, systems and/or organizations, and the early examination of soldier interface and missions. Finally, simulation is used to help determine system performance and safety. (For more information, refer to the Acquisition Modeling and Simulation Master Plan available via the OUSD-AT&L website at: www.acq.osd.mil/se/docs/AMSMP_041706_FINAL2.pdf.)

f. Training

M&S has been instrumental in preparing personnel for deployment to recent combat operations through the significant growth of "the ability to link together disparate virtual simulators and integrate constructive simulations to achieve complex mission environments" (United States Air Force [USAF], 2010). Through the delivery of integrated live, virtual, and constructive training environments that support personnel and mission rehearsal requirements, predeployment training exercises, and mission rehearsals, M&S has helped to ensure that personnel being deployed are trained and ready. The United States Marine Corps has fully embraced the use of simulations in training, and in an article on the Defense Transformation website (http://www.defense.gov/transformation/), one article states:

> "Live-fire training events come with a high price tag in the form of money, risks, logistics and time," said Jilson. Because of this, tactical decision-making simulations is one of the resources available to provide the same effective training without the risks. Research has also shown that there is effective training transfer when simulations are used to augment live training. In other words, when they are taught a new skill, they learn that skill better. (Bohanner, 2004)

g. Planning

The planning community utilizes a system of systems approach for examining long-term issues, which range from determining our next warfighting capabilities requirements, to examining the total force manpower. The assessment made by the Navy Modeling and Simulation Office (NMSO) is that "many of the emerging modeling technologies will play a key role in this new analytical realm, including agent-based models, computational social science, artificial societies, and behavior models" (NMSO, 2009). Figure 9 depicts the hierarchal relationship between models, and lists some of the models employed by the Navy for the respective hierarchies.

Figure 9. Hierarchy of Combat Models and Corresponding Navy Simulation Systems (From NMSO, 2009)

4. Related Studies

> If the relationships that compose a model are simple enough, it may be possible to use mathematical models to obtain exact information on questions of interest.
>
> —Law & McComas, (2001)

While there are no specific papers that discuss the paradigm of "simulation being a method of last resort," or when specifically to choose a simulation over an analytical solution, there are numerous papers and texts that identify situations in which the authors believe simulation should be utilized.

Averill Law (2007), in his textbook entitled *Simulation Modeling and Analysis*, provides the following list of applications of simulation:

- Designing and analyzing manufacturing systems
- Evaluating military weapon systems or their logistics requirements
- Determining hardware requirements or protocols for communication networks
- Determining hardware and software requirements for a computer system
- Designing and operating transportation systems such as airports, freeways, ports, and subways
- Evaluating designs for service organizations such as contact centers, fast-food restaurants, hospitals, and post offices
- Reengineering of business processes
- Analyzing supply chains
- Determining ordering policies for an inventory system
- Analyzing mining operations

When Law discusses analytical solutions versus simulation, he states that model complexity is the primary factor when deciding which approach to use. If the relationships between variables within the model are simple enough, Law states it is often possible to find an analytical solution. Law conditions this with the fact that some analytical solutions require vast computing resources, so while it may be possible to find an exact analytical solution, it may not be feasible based on available resources.

Pradeep Bhatia, of the Guru Jambheshwar University of Science and Technology in Hisar, India (http://www.gjust.ac.in/) provides some advantages and disadvantages to the use of simulation, along with recommendations on when to choose simulation as the primary option of exploration in his course notes for his class "System Simulation and Modeling." Bhatia (2010) states that the following are the advantages and disadvantages to simulation:

> **Advantages**: Simulation arbitrary models complexity, circumvents analytically intractable models, facilitates what-if and sensitivity analyses, and building a model can lead to system improvements and greater understanding that can be used to verify analytic solutions.
>
> **Disadvantages**: Simulation provides only estimates of the solution, only solves one parameter at a time, can take a large amount of development and/or computer time ("simulation as a last resort"). Don't use computer simulation if a common-sense or analytical solution is available, or if resources are insufficient, or if simulation costs outweigh benefits.

While we can agree with the advantages Professor Bhatia has listed, we cannot agree with the concept that simulation "only solves one parameter at a time" (Bhatia, 2010). The belief in this concept is perhaps one of the reasons the "last resort" paradigm has continued to plague the OR community. Through a proper design of experiments (DOE), one can search for robust solutions to problems, and examine the simultaneous effects of thousands of input variables concurrently (Kleijnen, Sanchez, Lucas, & Cioppa, 2005).

Professor Andrew Loerch of George Mason University (http://www.gmu.edu/) also addresses how one should choose between an analytical solution and a simulation in his course notes for OR540, Management Science (http://classweb.gmu.edu /aloerch/OR540.htm). In his notes, Professor Loerch states that in order to use a simulation to provide an optimal solution, one needs to analyze the output, make changes to the system, and then run the simulation again to see if an improvement has been achieved. He further states that one must weigh the advantages and disadvantages of using simulation, and use simulation only if the advantages outweigh the disadvantages.

Of the advantages Professor Loerch states, two are extremely important in the realm of military application of simulation. First, Professor Loerch (2001) states that simulation is "not subject to so many assumptions." Analytical solutions often require multiple assumptions to be made in order to find a solution (e.g., normality, independence, memorylessness, deterministic, linear, stationary, and homoscedasticity). One distinct benefit of simulation is that one can often reduce the number of or change the assumptions, which will allow for data outliers to enter the system. These outliers may identify critical problems or shortfalls within the system that would never have been seen using an analytical solution. Perhaps more importantly, Professor Loerch states that simulation results are "easy for [a] decision maker to understand" (2001).

With the premise that "unplanned, hit-or-miss course of experimentation with a simulation model can often be frustrating, inefficient, and ultimately unhelpful," Kelton (2000) provides a basic understanding on what is required in a simulation in order to maximize the benefit from it in his paper "Experimental Design for Simulation." This paper, presented at the 2000 Winter Simulation Conference, discussesfive key components that need to be addressed when developing a simulation:

- What model configuration should you run?
- How long should the run be?
- How many runs should you make?
- How should you interpret and analyze the output?
- What's the most efficient way to make the runs?

Professor Kelton (2000) then continues to educate his readers on the importance of utilizing carefully planned simulation studies to avoid "an undue amount of computational effort or (more importantly) your time."

B. RESEARCH QUESTIONS

This research seeks to answer the question as to when simulation should (and should not) be used to gain insight into a problem. It further seeks to identify the benefits one receives through a simulation approach to a problem, compared to the benefits of an analytical solution to the same problem.

C. BENEFIT OF THE STUDY

The primary benefit of this study is to demonstrate that the adage of simulation being a method of last resort is no longer valid. This paper will illustrate the benefits of using simulation to gain insight into complex problems and the adaptability to examine different assumptions of the input variables. Finally, the paper will highlight the responsibilities of the simulationist, analyst, and decision maker, which are inherent to a successful simulation study.

D. THESIS ORGANIZATION

The remainder of this thesis is organized into the following chapters. Chapter II provides an in-depth analysis of the analytical approach used by Parlar and Sharafali in their 2008 paper "Dynamic Allocation of Airline Check-In Counters: A Queuing Optimization Approach."

Chapter III begins with a discussion on the methodology employed by simulationists when approaching a problem, followed by a brief introduction to the Simio software suite. With the basics of the simulation process explained, they are then illustrated through the development of a simulation created using as close to the criteria and assumptions made by Parlar and Sharafali as possible. The chapter concludes with a demonstration of how simulation can often provide more insight to a problem through the ability to easily explore different assumptions and distributions with no change to the core model.

The research is concluded in Chapter IV, where the significant findings to the research questions are posed. Additionally, Chapter IV provides recommendations for future improvements and extensions of this research in order to demonstrate how the use of simulation should be one of "first resort" for many complex, real-world problems.

II. ANALYSIS OF AN ANALYTICAL SOLUTION TO A COMPLEX PROBLEM

> Because our model is analytical and more realistic, the optimization results are on firmer ground than those based on simulation.
>
> —Parlar and Sharafali, (2008)

A. BACKGROUND

Mahmut Parlar and Moosa Sharafali published a paper entitled "Dynamic Allocation of Airline Check-In Counters: A Queuing Optimization Approach" in *Management Science* in August 2008. In this paper, the authors proposed an analytical method to optimize the management of airline check-in counters. They claim that while others have "resorted to simulation," they found that "this complicated problem is amenable to analytical treatment."

The authors were motivated to write this paper after witnessing the demands placed on the counter management staff to meet their customer-service criteria, security requirements, and service efficiency standards in various international airports. They claim that there should be two phases in their research, of which they are focusing only on the first. Parlar and Sharafali's focus was to study "the queuing and statistical analysis phase to help determine the optimal number of counters needed for each flight over time to minimize a certain expected cost function (while implicitly achieving a desired level of customer service)," (Parlar & Sharafali, 2008).

The authors saw two distinct benefits to airline-counter management as a result of their research. First, their methodology, through the use of a dynamic program, would enable counter management staff to make optimal decisions on manning the counters to minimize total cost. Second, the methodology could also be utilized to ensure that the airline is meeting a quality of service standard.

Our intent through this discussion is not to fully define and explain the methods used by Parlar and Sharafali, but to provide a simplified framework with which their

assumptions can be seen in their analytical context. For a detailed explanation on their methodology, it is best to refer to their paper.

B. ASSUMPTIONS

> Such an assumption would also destroy the Markovian nature of the process and make the model intractable.
>
> —Parlar & Sharafali, (2008)

> We have always assumed that the lifetimes and the service times are both exponential, and with this assumption we derived analytic expressions for quantities of interest (the most important being the transient solution of the conditional probability.
>
> —Parlar & Sharafali, (2008)

Assumptions underpin all models; in practice, to make an analytical solution possible, many assumptions must often be made to constrain the problem to something that is solvable. In Parlar and Sharafali's case, it was the authors' goal to solve the problem, their result was the solution to a mathematical exercise, with little relevance to the original problem (Type III error). With the goal to develop an analytical solution, the authors had to continually make assumptions until they were in a position to use an analytical process.

In their paper, the authors explicitly stated some of the assumptions they made in the model development. However, while some of the assumptions were stated, simplifying assumptions were not addressed, and it is important to recognize all assumptions made in the model—not just the stated ones.

1. Passengers

To begin, let us examine the assumptions made concerning the passengers. The first assumption is that all passengers for a specific flight will arrive at the window accounted for by the model (e.g., a 3-hour window prior to flight). This assumption is in direct conflict with reality. The problem of no-shows for any given flight has been a problem that has vexed the airline industry since its inception, as each empty ticket is a loss of potential revenue. In one recent study, the mean no-show rate for a given flight

was approximately 10% (Lawrence, Hong, & Cherrier 2003), which is why airlines attempt to factor in the expected number of no-shows when overbooking a flight.

The next assumption is that all passengers arrive as individuals. From anecdotal evidence alone, we know this to be categorically false. This has several significant effects on their optimal solution. First, the calculated arrival rates for the subintervals do not take group arrival into consideration. Second, the service time is based on assisting an individual, vice a group of people who are traveling together (e.g., a husband and wife). Another key assumption is that all passengers have the same complexity and needs. This, too, each of us knows to be false based on personal experience. There are times when scheduling issues, seat issues, or connection issues take place that make the initial check-in process more complex. In addition, passengers arrive according to a specific profile, which must be known in advance. This profile is used to determine the length of the subintervals and the arrival rates $\left(\hat{\lambda}_k\right)$ for each subinterval.

The calculation and use of $\hat{\lambda}_k$ for each subinterval is also suspect. Their calculations are based on the procedure given in Basawa and Prakasa Rao (1980). This procedure calculates a constant arrival rate over a subinterval. See Parlar and Sharafali (2008) (when referring to the model and assumptions made by Parlar and Sharafali, it shall be henceforth referred to as the Parlar Model for brevity) for amplification.

$$V_{T_k} \equiv \sum_{i=0}^{n-1} x_i \left(\tau_{i+1} - \tau_i\right) + x_n \left(T_k - \tau_n\right)$$
$$\hat{\lambda}_k = n_k / V_{T_k}$$

Through the use of the equations above, Parlar and Sharafali provide a numeric example, as shown in Table 2 (from this table, each period k is one hour in length). To see a detailed definition of these variables, refer to Parlar and Sharafali (2008).

Period k	No. of Passengers Arrived (n_k)	Arrival Times $[\tau_1 < \cdots < \tau_n]$(hrs.)	$\hat{\lambda}_k$
1	4	[0.32, 0.34, 0.42, 0.47]	0.31
2	6	[1.15, 1.46, 1.47, 1.58, 1.93, 1.96]	0.69
3	5	[2.11, 2.44, 2.57, 2.71, 2.87	1.83

Table 2. Numerical Example of Estimated Arrival Rates (From Parlar & Sharafali, 2008)

Given that these arrival rates are constant over each 1-hour period, one would expect that the projected number of arrivals would be able to be calculated. In this case, there were 15 arrivals scheduled; however, using the calculated arrival rates, one cannot match the expected number of arrivals with the number that actually arrived.

Finally, the authors assumed that all passengers arriving are traveling in the same class. This is instrumental in their calculation of the passenger delay cost, as the cost is set at a fixed value for all passengers. This, too, does not reflect the reality of most airline traffic. It only applies when one is addressing a very minor number of instances where there is no class differentiation on price of seat, or significance of the passengers.

2. Service

The way in which the customers are served is the next set of assumptions that are examined. First, over any given epoch, a constant number of counters are available. The number of open counters can only change at the beginning of each period (k), which is based on the number people who have arrived and the number of people who have arrived and already completed service. The times at which these decisions are made are based on the history of previous arrivals on which the arrival rates have already been calculated. Thus far, the conditions set are not too unrealistic. However, when the service rate is examined, it is another story. First, the authors assume that the service rate for a given counter (μ) is the same for each counter. Second, the service rate increases as a function of the number of people who have not been served, and the number of open counters. This presents a major issue with the analytical solution, as it is not reasonable

that service rates increase linearly as a function of the number who have arrived and not yet had service completed. Table 3 illustrates service rate (as a function of passengers per unit time) with this assumption.

		Number of Passengers Arrived Who Have Not Completed Service									
		1	2	3	4	5	6	7	9	9	10
Number of Open Counters	1	5	10	15	20	25	30	35	40	45	50
	2	10	20	30	40	50	60	70	80	90	100
	3	15	30	45	60	75	90	105	120	135	150
	4	20	40	60	80	100	120	140	160	180	200
	5	25	50	75	100	125	150	175	200	225	250
		Service Rate (Passengers/Unit Time)									

Table 3. Service Rate Table Based on the Parlar Model

In addition to an impossible increase in rate of service, the authors assume an exponential service time distribution in order to make the model mathematically tractable. The fact that the mode of the exponential probability density function is equal to zero, one can reasonably argue against the validity of the “most likely” service time is zero assumption. The authors did provide a discussion regarding the potential use of Erlang service times, but concluded that “such a generalization would probably require a computational approach (rather than an analytic approach as used in this paper) due to the resulting high dimensionality” (Parlar & Sharafali, 2008). This statement illustrates that the author’s goal was to provide an analytical solution vice a solution which actually addresses the problem (Type III error).

3. Costs

The goal of the authors was to provide an optimal solution that is based on minimizing cost to the airline; thus, the assumptions made regarding cost play a significant role in the development of their solution.

The first cost is the Passenger Delay Cost (C_w). The authors chose to use C_w=\$40, which they claim is close to the value of the Federal Aviation Administration’s (FAA) estimate for business class customers. As mentioned previously, all passengers are considered to be in the same class. The total expected cost for the passengers during

each period is based on the calculation for the expected wait time during each period. This value is determined by the authors as follows (in passenger hours):

$$W_k(c_k) = \left[\frac{\lambda(N-m_k)}{\lambda - c_k\mu} + (m_k - n_k)\right]\frac{e^{-c_k\mu t_k} - e^{-c_k\mu t_{k+1}}}{c_k\mu} + \frac{(N-m_k)}{\lambda - c_k\mu}\left(e^{-\lambda t_{k+1}} - e^{-\lambda t_k}\right).$$

It is important to note that in this equation, the arrival rates are based on the previous estimates, and the service rate is affected by the number of open counters and the number of passengers who have arrived, but not completed service. Thus, any problem with the assumptions has a direct impact on the optimal solution they calculate. Another important item to note regarding this cost is that the cost is only attributed to the expected delay that the passenger experiences while waiting in the queue and the service time itself. The time to walk between their arrival point and the security checkpoint are not accounted for.

The next cost assumed is the Check-In Counter Operating Cost, (C_s). Here the authors chose to use a value of $C_s = 60$. The authors based this cost on their "best guess" of associated costs, as no real analysis had been previously done to provide a cost estimate for the assumptions made in their counter model. Their estimate is based on the cost of the employee, use of the telephone, counter space, restrooms, and etcetera. The interesting part of this assumption is that the cost is a deterministic cost, based solely on the number of counters open during a specific period. This means that the cost is only incurred when the counter is manned.

The cost for each interval is based on the number of counters open during the interval and the expected wait time and cost of the passengers:

$$g_k(c_k) = g_k(c_k, m_k, n_k) = C_w W_k(c_k) + C_s(t_{k+1} - t_k)c_k.$$

The third cost taken into account is the Aircraft Delay Cost, (h). The authors chose $h = \$20$ as the value to use in the optimization. This represents the cost incurred by the airline for every passenger not cleared when the counter is expected to close. The authors assumed that this would most likely cause a delay in the departure of the aircraft.

The authors used data from the European Organization for the Safety of Air Navigation (EOSAN), which provides a cost estimate due to delays in transportation by source in cost per minute values. The authors assumed that the delay would be due to people not checked in, and took the average duration of delay and divided it by the number of passengers not checked in. While some airlines may have sufficient slack in their schedule that they may wait for a passenger to arrive, one can safely say that it would be the exception to the norm if it occurred.

C. INTERPRETING THE SOLUTION

> The error of the third kind is the error committed by giving the right answer to the wrong problem.
>
> —Kimball, (1957)

After Kimball provided the definition of the Type III error as quoted above, he went on to say "In defining it this way we are allowing the statistician the benefit of the doubt by rejecting the possibility that he would give the wrong answer to the wrong question" (2008). In the case of Parlar and Sharafali's (2008) paper, one must wonder if this is an example of a Type III error.

Parlar and Sharafali (2008) utilized a dynamic program written in Maple to calculate their optimal solution. The optimal solution they provide is a table that indicates the number of counters that should be open during a specific epoch, given the number of people who have arrived and the number of people who have already been served. Table 4 lists the parameters that were used as an example to compute their output:

T	K	N	$c_k^{\min}$	$c_k^{\max}$	$\hat{\lambda}_1$	$\hat{\lambda}_2$	$\hat{\lambda}_3$	μ	C_w	C_s	h
1	3	10	1	5	0.58	1.60	2.74	5	40	60	20

Table 4. Parameters for Numerical Example (From Parlar & Sharafali, 2008)

Based on the values above, the dynamic program calculated the transient probabilities and the expected number of passengers in the system. Based on those values, the dynamic program then calculated the minimum expected cost to go from the beginning of the subinterval until the final time using the following expression:

$$V_k(m,n) = \min_{\substack{c_k^{\min} \le c_k \le c_k^{\max} \\ c_k(N,N)=0, V_k(N,N)=0}} \left[g_k(c_k) + \sum_{i=m}^{N} \sum_{j=n}^{i} p_{m,n}^{c}(i,j,t_{k+1}-t_k) V_{k+1}(i,j) \right].$$

The output of the dynamic program was consolidated into Table 5.

	n										
c_1^*, c_2^*, c_3^*	0	1	2	3	4	5	6	7	8	9	10
m = 0	1,1,4										
m = 1	1,1,4	1,1,4									
m = 2	1,1,4	1,1,4	1,1,4								
m = 3	1,1,3	1,1,3	1,1,3	1,1,3							
m = 4	1,1,3	1,1,3	1,1,3	1,1,3	1,1,3						
m = 5	1,1,3	1,1,3	1,1,3	1,1,3	1,1,3	1,1,3					
m = 6	1,1,3	1,1,3	1,1,3	1,1,3	1,1,3	1,1,3	1,1,3				
m = 7	2,2,3	1,1,3	1,1,3	1,1,3	1,1,3	1,1,3	1,1,2	1,1,2			
m = 8	2,2,3	2,2,3	1,1,3	1,1,3	1,1,3	1,1,2	1,1,2	1,1,2	1,1,2		
m = 9	2,2,3	2,2,3	2,2,3	2,2,3	1,1,2	1,1,2	1,1,2	1,1,2	1,1,2	1,1,1	
m = 10	2,2,3	2,2,3	2,2,3	2,2,3	2,2,3	2,2,2	2,2,2	1,1,2	1,1,2	1,1,1	0,0,0

Table 5. Optimal Number of Counters to Open for Any $c_k^*(m,n)$ for Any State (m,n) at $k = 1,2,3$ (From Parlar & Sharafali, 2008)

To interpret the output, one must know the epoch, number of arrivals (m), and the number of passengers who have completed service (n) at the end of a given epoch. For example, if at the end of epoch 2, six passengers have arrived $(m = 6)$ and four of those passengers have completed service $(n = 4)$, three counters need to be opened for epoch 3 $(c_3^* = 3)$.

Despite receiving a copy of the dynamic program code, we were unable to reproduce the exact table above based on the same inputs. Additionally, it is important to note that Parlar and Sharafali provide a disclaimer on the use of their "optimal" solution when they state that "that our intention here is just to highlight the usability of our model rather than the use of very realistic estimates as input into our models" (Parlar & Sharafali, 2008).

THIS PAGE INTENTIONALLY LEFT BLANK

III. THE SIMULATION APPROACH

> There are some serious misunderstandings concerning the nature of simulation and its ease of employment. The truth of the matter is that there's no such thing as 'simple simulation.' It's a myth often inadvertently perpetuated by manufacturers of simulation software and professors who want their students to believe they're learning an easier alternative to tools like linear programming.
>
> —Keller, Harrell, and Leavy, (1991)

A. HOW A SIMULATIONIST WOULD APPROACH THE PROBLEM

Throughout operations research textbooks (e.g., Hillier & Lieberman, 1986), simulation textbooks (e.g., Law & Kelton, 2000), and papers, the methodology used to develop a solution to a given problem is consistent. The DoD's MSCO for Verification, Validation, and Accreditation (VV&A) cite the seven-step approach (developed by Law and Kelton) to successful simulation modeling in the document entitled "A Practitioner's Perspective on Simulation Validation," available via their website at http://vva.msco.mil/Key/key-pr.pdf. It is through this process that the simulation for this paper is developed. The logical flow of this seven-step approach is illustrated in Figure 10.

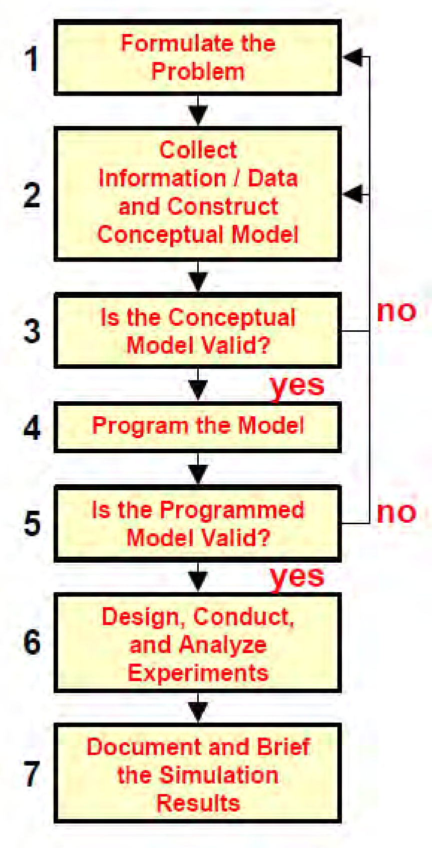


Figure 10. Seven Step Approach to Successful Simulation and Modeling, (From MSCO, 2001)

1. FORMULATE THE PROBLEM

Essential to formulating the problem is the identification of the problem itself. With the problem identified, it is then essential to determine what the overall objective of the study is and which specific questions need to be answered. These are essential to know in the very beginning of the process in order to ensure that not only the correct data are collected, but the correct output mechanisms are incorporated. These specific questions and objectives will shape the scope of the model and the conditions under which the simulation will be run.

2. COLLECT INFORMATION/DATA AND CONSTRUCT CONCEPTUAL MODEL

> The most successful procedure for model building has been to work from simple and small to the larger and more complex.
>
> —Ancker, (1995)

With the problem formulated and the specific questions and objectives specified, the next step is to collect the information necessary to create a simulation that operates in the same manner as the airline-counter system being studied. The data collected are used to program the model parameters and the probability distributions of events in the system (e.g., arrival times of passengers, number of passengers who travel together, and the service times of passengers).

Essential to this process is ensuring that any assumptions made for the system are documented, along with any algorithms used when developing a conceptual model. There are several factors that help determine what level of detail should be represented within the model. Some of these are constraints on model complexity due to the computer system being utilized, the cost of programming, and the amount of time it takes. Others are to ensure there is sufficient detail to be able to address the credibility of the model. For example, if the objective is to determine how many service counters to open at an airport to minimize the delay passengers experience, while also minimizing the operating costs, one can model the arrivals from the time they enter the airport to the time they have completed checking in. Modeling their method of arrival, highway or parking delays, or their passage through the security checkpoints are all irrelevant when it comes down to answering the specific question posed in the formulation process.

3. IS THE CONCEPTUAL MODEL VALID?

Once the conceptual model has been developed, it is critical that a structured walk-through is conducted to verify its validity. This walk-through needs to be conducted with people who are knowledgeable about the system to ensure that any erroneous assumptions on operations are addressed, and along with the analysts, to ensure that the required metrics are being output. This is an iterative process, so if there are any

issues, it is important to address them, redesign the model (or reformulate the problem if it was not being understood correctly), and then check the model for validity again.

4. PROGRAM THE MODEL

> For any given study, in selecting the tools to use, analysts must weigh several competing attributes, such as the models' ease of use, agility, transparency, reproducibility, and realism.
>
> —Lucas and McGunnigle, (2003)

As pointed out by Lucas and McGunnigle (2003), choosing the right tool is essential in constructing the simulation model. The tool that one picks is based on availability, knowledge, cost, and capabilities. Some models can be implemented in commercial programs such as Simio and ARENA, whereas others are better suited to be done in a programming language such as C++ or JAVA. Regardless of the software used to implement the model, the limitations of the software and how those limitations affect the capability of the software to implement the model as designed need to be known, understood, and accounted for in the design and in the analysis portion.

5. IS THE PROGRAMMED MODEL VALID?

Whether a solution is provided through an analytical model or a simulation, one must be able to validate the results. The output from the model needs to be reviewed by subject matter experts to determine whether or not the results are consistent with the understanding of the system.

There has been a substantial volume of information published on the VV&A process. Osman Balci, in his paper for the 1997 Winter Simulation Conference, published a very succinct taxonomy of the verification and validation techniques for both conventional and object-oriented simulation models. Figure 11 is the taxonomy for his conventional simulation models.

V&V Techniques for Simulation Models

Informal

- Audit
- Desk Checking
- Documentation Checking
- Face Validation
- Inspections
- Reviews
- Turing Test
- Walkthroughs

Static

- Cause-Effect Graphing
- *Control Analysis*
  - Calling Structure Analysis
  - Concurrent Process Analysis
  - Control Flow Analysis
  - State Transition Analysis
- *Data Analysis*
  - Data Dependency Analysis
  - Data Flow Analysis
- Fault/Failure Analysis
- *Interface Analysis*
  - Model Interface Analysis
  - User Interface Analysis
- Semantic Analysis
- Structural Analysis
- Symbolic Evaluation
- Syntax Analysis
- Traceability Assessment

Dynamic

- Acceptance Testing
- Alpha Testing
- Assertion Checking
- Beta Testing
- Bottom-Up Testing
- Comparison Testing
- *Compliance Testing*
  - Authorization Testing
  - Performance Testing
  - Security Testing
  - Standards Testing
- Debugging
- *Execution Testing*
  - Execution Monitoring
  - Execution Profiling
  - Execution Tracing
- Fault/Failure Insertion Testing
- Field Testing
- Functional (Black-Box)Testing
- Graphical Comparisons
- *Interface Testing*
  - Data Interface Testing
  - Model Interface Testing
  - User Interface Testing
- Object-Flow Testing
- Partition Testing
- Predictive Validation
- Product Testing
- Regression Testing
- Sensitivity Analysis
- *Special Input Testing*
  - Boundary Value Testing
  - Equivalence Partitioning Testing
  - Extreme Input Testing
  - Invalid Input Testing
  - Real-Time Input Testing
  - Self-Driven Input Testing
  - Stress Testing
  - Trace-Driven Input Testing
- *Statistical Techniques*
- *Structural (White-Box)Testing*
  - Branch Testing
  - Condition Testing
  - Data Flow Testing
  - Loop Testing
  - Path Testing
  - Statement Testing
- Submodel/Module Testing
- Symbolic Debugging
- Top-Down Testing
- Visualization/Animation

Formal

- Induction
- Inductive Assertions
- Inference
- Lambda Calculus
- Logical Deduction
- Predicate Calculus
- Predicate Transformation
- Proof of Correctness

Figure 11. Taxonomy of Verification and Validation Techniques for Conventional Simulation Models (From Balci, 1997)

6. DESIGN, CONDUCT, AND ANALYZE EXPERIMENTS

> Unfortunately, few simulation practitioners seem to be aware of the additional insights that can be gleaned by effective use of designs.
>
> —Kleijnen et al., (2005)

Understanding how to properly design a series of experiments for the maximum effectiveness is critical for both the stakeholder and the analyst. Through a robust DOE, one can gain a better understanding of the system and find more robust solutions than if one limits the design to looking only for an optimal solution. In the paper entitled "A User's Guide to the Brave New World of Designing Simulation Experiments," Kleinjnen et al. (2005) attempt to change the mindset of simulation practitioners and researchers so that the DOE process is considered instrumental in any simulation study.

When a system is complex with many variables, it becomes a significant challenge to run a full-factorial design. Through a proper DOE and utilizing tools such as efficient nearly orthogonal and space-filling Latin hypercubes (see Cioppa & Lucas, 2007), one can maximize the efficiency and ability of the analysts to "examine multiple factors within the simulation through fitting models with main, quadratic, and interaction effects with nearly uncorrelated estimates of the regression coefficients for the linear effect terms" (Cioppa & Lucas, 2007).

7. DOCUMENT AND BRIEF THE RESULTS

In order for the simulation to be useful to the person who requested it in the first place, the results need to be presented in such a manner that the true implications of the analysis are unambiguous. Simply stating the mean values of some of the parameters of interest is insufficient when presenting the results of the simulation to the stakeholders. It is imperative that the analyst not only provide the means, but the confidence intervals and the conditions under which those means were ascertained. Then, if sensitivity analysis was conducted, present that information as well, followed by the identification of possible improvements to remove bias, increase precision (fidelity), or other factors needed for better sensitivity analysis.

B. THE SIMIO SOFTWARE

The software chosen to implement a simulation version of the Parlar Model is Simio (www.simio.com). Simio (Simulation Modeling framework based on Intelligent Objects) provides a graphical, object-oriented modeling framework that allows for both continuous and discrete time events, as well as supporting an event, process, object, and agent modeling view. Figure 12 shows the graphical user interface of Simio, which allows for representation in both two and three dimensions.

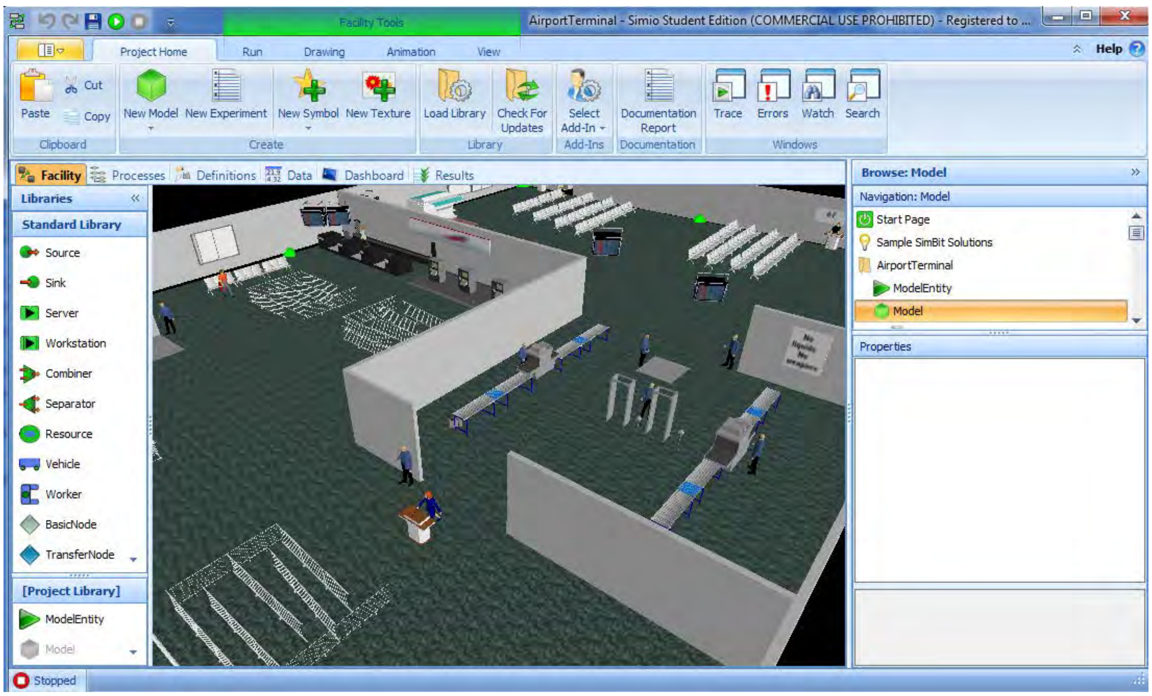


Figure 12. Simio Model View of the Airport Terminal Example Scenario

With a built-in analytical tool, the "quick look" of the simulation data is extremely convenient. It allows for the user to verify the outputs of the model during the course of the model development without having to export the results and interpret the results using additional software. The software allows for a box plot, histogram, individual observations, confidence intervals, and a means line to all be depicted on the same graph. Figure 13 illustrates this functionality utilizing two different scenarios of the same model.

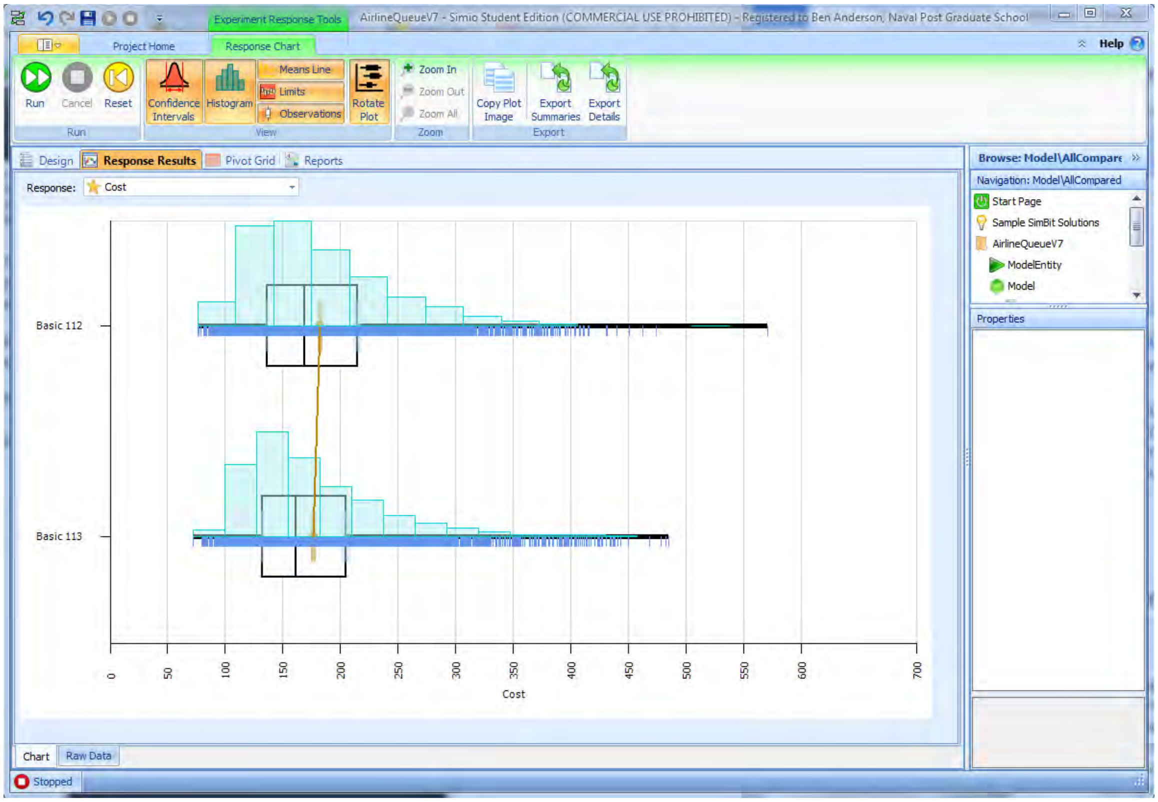


Figure 13. Simio Response Results Display—Depicting Two Different Scenario Evaluations of the Same Model

Taking full advantage of multiple core processors, Simio can run multiple scenarios concurrently, each with separate streams of random numbers. This greatly reduces the total run time on an average computer system (Simio, 2011). Further, leveraging increased graphics capability, Simio allows for both two-dimensional and three-dimensional graphical representation of the system. Through the use of standardized coding, Simio has a direct interface to the Google 3D Warehouse, allowing for the simulation to be modeled visually as accurately as possible. For some systems (and decision makers), the visual component provides insight that the analytical component might not make apparent.

Simio also provides an "Experimentation Mode," which allows for a DOE to be run, allowing as many scenarios and replications of each scenario run as desired, with the output of all readily available to either view within Simio or be exported to a comma-separated values (CSV) file for further analysis.

For detailed information regarding the Simio software one should refer to the *Simio Reference Guide*, available at www.simio.com/academics/simio-academic-resources.htm.

C. SIMULATION MODEL BASED ON THE PARLAR MODEL

1. Formulation, Data Collection, and Conceptual Model

To follow the steps as discussed previously, the first step in building our model is to identify the problem. In this case, the problem is to ascertain the validity of the analytical solution presented by the Parlar Model. With the problem identified, the next step is to formulate the problem. Here, we will attempt to replicate the same assumptions and conditions and time frames as done by Parlar, hence the data, distributions, and associated problem parameters should be the same in the simulation model as in the analytical model.

Using the same parameters as Parlar and Sharafali (2008) (see Table 6), we attempt to replicate the Parlar Model as accurately as possible. A minimum of $c_k^{\min} = 1$ counters must be open, while no more $c_k^{\max} = 5$ counters being open during a given epoch. With $K = 3$ epochs and $T = 1$ hour, each epoch is 20 minutes in length. All $N = 10$ passengers will arrive during that time frame, with the estimated arrival rate of $\hat{\lambda}_k$ per epoch. These passengers will be processed at a mean rate of $\mu = 5$ (passengers per hour). The cost of passenger delay is $C_w = 40$ per hour, while the cost of delaying the aircraft for every passenger not checked in by is $h = 20$ per passenger. The cost of operating the counters is $C_s = 60$ per counter, per hour.

T	K	N	$c_k^{\min}$	$c_k^{\max}$	$\hat{\lambda}_1$	$\hat{\lambda}_2$	$\hat{\lambda}_3$	μ	C_w	C_s	h
1	3	10	1	5	0.58	1.60	2.74	5	40	60	20

Table 6. Parameters for Numerical Example (From Parlar & Sharafali, 2008)

During the development of the conceptual model, a problem was identified with the given arrival rates per epoch. We were unable to produce a system using the above

rate parameters such that exactly $N = 10$ passengers arrive within the $T = 1$ hour period. This brought into question the validity of the $\hat{\lambda}_k$'s. A simple experiment was done in order to ascertain whether or not these were valid values. Given the following:

- $\hat{\lambda}_k$'s are known,
- $\hat{\lambda}_k = n_k / V_{T_k}$, and
- $V_{T_k} \equiv \sum_{i=0}^{n-1} x_i \left(\tau_{i+1} - \tau_i\right) + x_n \left(T_k - \tau_n\right)$.

Utilizing Microsoft EXCEL and the SOLVER add-in, $\hat{\lambda}_1$ was calculated for $n_1 = \{1,2,3\}$ to minimize or maximize the value of $\hat{\lambda}_1$ in order to determine if $\hat{\lambda}_1$=0.58 was feasible. Based on the calculations shown in Table 7, the value for $\hat{\lambda}_1$=0.58 is not feasible. Thus, it is apparent that the values provided in the paper were either not calculated in accordance with the method indicated, or there was an error during the publication process where incorrect values were substituted.

x_i	τ_{i+1}	τ_i	$x_i(\tau_{i+1}-\tau_i)$	V_{Tk}	n_k/V_{Tk}	Solved
Three Arrivals						
10.0000	0.3333	0.0000	3.3333	3.3333	0.9000	MIN
9.0000	0.3333	0.3333	0.0000			
8.0000	0.3333	0.3333	0.0000			
7.0000	0.3333	0.3333	0.0000			
Two Arrivals						
10.0000	0.3333	0.0000	3.3333	3.3333	0.6000	MIN
9.0000	0.3333	0.3333	0.0000			
8.0000	0.3333	0.3333	0.0000			
One Arrival						
10.0000	0.0000	0.0000	0.0000	3.0000	0.3333	MAX
9.0000	0.3333	0.0000	3.0000			
Cells in blue represent the $x_n(T_k-\tau_i)$ portion of function						

Table 7. Calculation of $\hat{\lambda}_1$ Based on the Parlar Model Parameters

In order to provide a simulation result as close as possible to the Parlar Model's conditions, the values of 5.22, 14.4, 24.66 (the original $\hat{\lambda}_k$'s multiplied by a factor of 9) produced an average of 9.8 arrivals over the 1-hour period. Runs during which 10 arrivals did not occur were omitted from analysis after the preliminary examinations.

2. Program and Validate Model

a. Basic Model

As the scope of the model in this instance is designed to specifically represent the same system as in the Parlar paper, the conceptual model was very easy to build. To program the model, Simio provides a great flexibility that programming in SIMKIT or developing the simulation in ARENA did not, as Simio has a very easy to learn (and use) graphical user interface (contrary to SIMKIT), and an installed experimentation, optimization, and analysis tool (contrary to ARENA). To start the model, a generic source, server, and sink are dragged and dropped from the Standard Library onto the Modeling Canvas (see Figure 14), then the objects are renamed to be representative of their respective functions of "Arrivals," "Counters," and "Security."

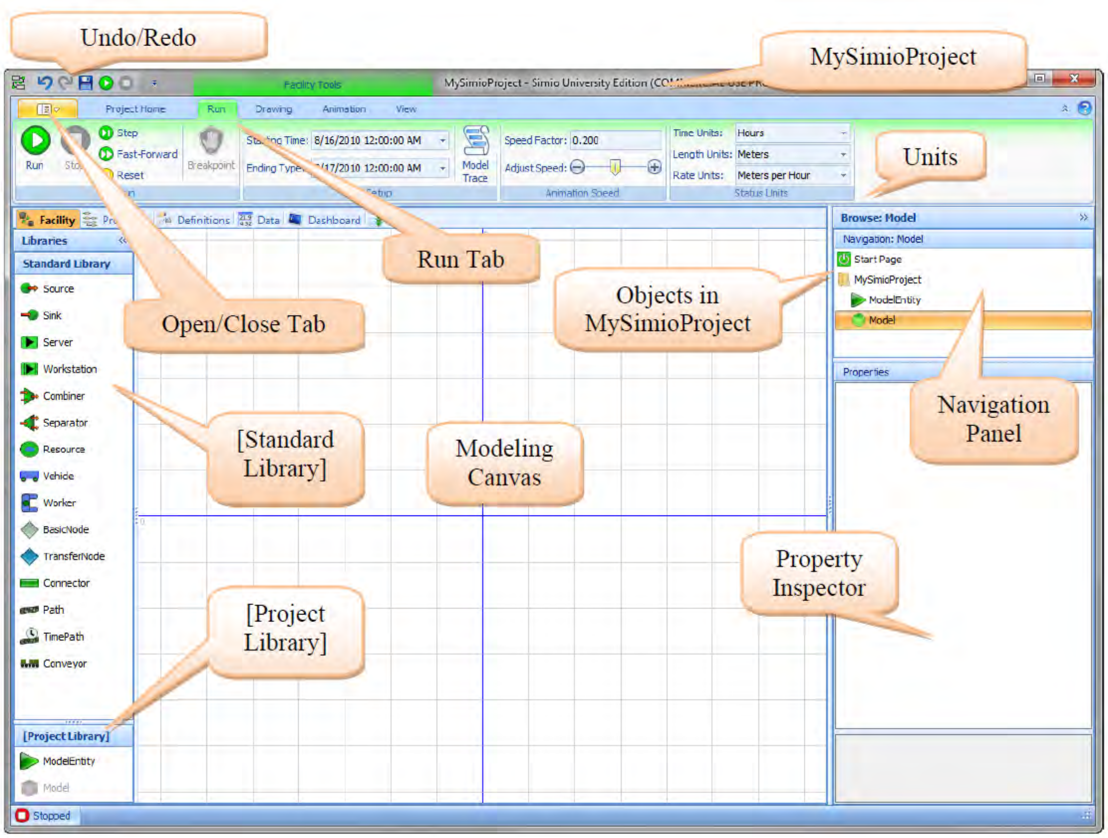


Figure 14. Simio User Interface when Creating a New Model (From Joines & Roberts, 2010)

As seen in Figure 15, each of these objects is connected via a "Connector" object (again taken from the Standard Library), which allows for a zero time transit between the objects (as assumed in the Parlar Model).

Figure 15. Basic Simio Implementaiton of the Parlar Model

As mentioned previously, Simio has an object-oriented architecture. Combining the object-oriented architecture with an easy to navigate graphical user interface, it is extremely easy for beginning users to quickly customize each of the objects in the Standard Library. By selecting an object on the Modeling Canvas, the properties of that object are displayed in the Property Inspector (see Figure 15). To establish some of the conditions of the Parlar Model, the "Arrivals" object must be modified. By selecting the "Arrivals" object, the properties of that object are then displayed (see Table 8). The first parameter to change is "Arrival Mode," by selecting the "Time Varying Arrival Rate." This allows for a rate table to be utilized in lieu of a single distribution, allowing for complex arrival rates to be modeled within the simulation, which is much more representative of reality. A portion of the simulation control is also embedded in the "Arrivals" object, as it will only produce a maximum of N arrivals (where N was previously defined as $N = 10$ by the Parlar Model).

Properties: Arrivals (Source)	
Arrival Logic	
Entity Type	**Passenger**
Arrival Mode	**Time Varying Arrival Rate**
Rate Table	**ArrivalRates**
Rate Scale Factor	1.0
Entities Per Arrival	1
Stopping Conditions	
Maximum Arrivals	**N**
Maximum Time	Infinity
Stop Event Name	
Table Reference Assignments	
State Assignments	
Add-On Process Triggers	
Advanced Options	
General	
Name	Arrivals
Description	Source of passengers arriving to ...
Public	True
Report Statistics	True
Size and Location	
Animation	

Table 8. Properties of the "Arrivals" Object

The Arrivals Rate Table (as seen in Table 9) allows for any number of time periods of constant length to be entered. Upon reaching the end of rate table, Simio loops back to the first row if the simulation is still running.

Name	ObjectType	DisplayName
Rate Tables		
ArrivalRates	Rate Table	ArrivalRates

Starting Offset	Ending Offset	Rate (events per hour)
Day 1, 00:00:00	Day 1, 00:20:00	5.22
Day 1, 00:20:00	Day 1, 00:40:00	14.4
Day 1, 00:40:00	Day 1, 01:00:00	24.66

Table 9. Arrivals Rate Table

With the parameters set for the "Arrivals" object, the "Counters" object requires settings to be established (see Table 10). First, the initial server capacity is set to a variable "E1SC" (epoch 1 Server Capacity). The processing time for each entity is determined through a random exponential draw, based on the variable "ServiceRate." Upon the model's initialization, the "EstablishInitialServiceCounterRate" process is called, and then upon processing each entity, the "UpdateServiceRate" process is called (see Figure 16). This process is included to mimic the Parlar assumption that the service rate at the counter will increase as a function of the number of people in line and being served. Table 10 shows the properties of the "Counters" object.

Properties: Counters (Server)

Process Logic	
Capacity Type	Fixed
Initial Capacity	**E1SC**
Ranking Rule	First In First Out
Dynamic Selection Rule	None
Transfer-In Time	0.0
Processing Time	**Random.Exponential(ServiceRate)**
Buffer Capacity	
Reliability Logic	
State Assignments	
Secondary Resources	
Add-On Process Triggers	
Run Initialized	**EstablishInitialServiceCounterRate**
Run Ending	
Entered	
Processing	**UpdateServiceRate**
Processed	
Exited	
Failed	
Repairing	
Repaired	
On Shift	
Off Shift	
Advanced Options	
General	
Animation	

Table 10. Properties of the "Counters" Object

The cost of the counters is tallied using a "Rate Tally Statistic." This allows for the rate at which cost is accrued (based on the number of open counters and

the counter operating cost) to happen in the background since that portion of the cost is not passenger dependent. The first time this process is called is on the initialization of the project. It also will run each time the “Counters” object capacity (which represents the number of open counters) changes (see Figure 16 and Table 11).

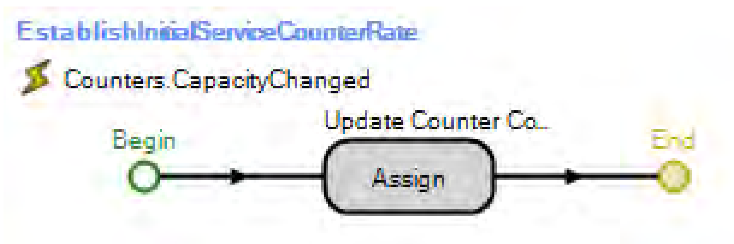


Figure 16. Update Counter Cost Rate Process

Properties: Update Counter Cost Rate (Assign Step Instance)	
Basic Logic	
State Variable Name	**TC_Counters.Rate**
New Value	**Cs*Counters.CurrentCapacity**
Assignments (More)	0 Rows
Advanced Options	
General	

Table 11. Assignment for Total Counter Cost Tally Statistic

A key assumption of the Parlar Model is that the service rate provided is dependent on the number of people in the system who have not completed service (see Figure 17). As one of the tests that will be run removes this assumption, two different assignments functions are utilized, depending on which case is being examined (see Table 12).

Figure 17. Update Service Rate Process

Basic Logic	
State Variable Name	**ServiceRate**
New Value	**60/(mu*(Counters.InputBuffer.NumberEntered-Security.InputBuffer.NumberEntered))**
Assignments (More)	0 Rows
Advanced Options	
General	

Basic Logic	
State Variable Name	**ServiceRate**
New Value	**60/mu**
Assignments (More)	0 Rows
Advanced Options	
General	

Table 12. Service Rate Calculations

Upon completion of service, the passengers arrive at the "Security" object. As the entities are destroyed, the first cost process is called (see Table 13 and Figure 18). The first portion of this process is to record the passenger delay cost in the variable assigned to track the total cost of passenger delay (see Table 14). The process then checks to see if all N passengers have arrived, and if so, closes the counters (see Tables 15 and 16).

Properties: Security (Sink)	
Process Logic	
Transfer-In Time	0.0
State Assignments	
Add-On Process Triggers	
Run Initialized	
Run Ending	
Entered	
Destroying Entity	**AddAllPassengerCosts**
Advanced Options	
General	
Animation	

Table 13. Properties of the "Security" Object

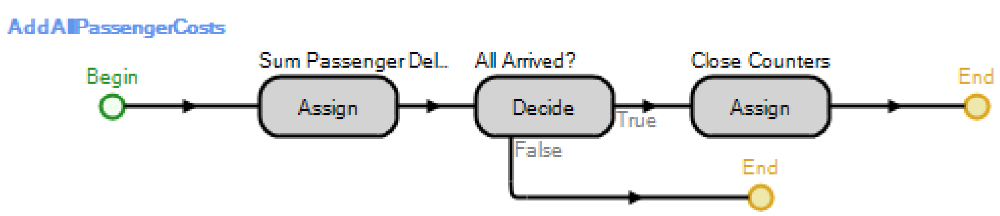


Figure 18. Add All Passenger Costs Process

Properties: Sum Passenger Delay (Assign Step Instance)	
Basic Logic	
State Variable Name	**TC_PassengerDelay**
New Value	**TC_PassengerDelay+Security.TimeInSystem.LastRecordedValue*Cw**
Assignments (More)	0 Rows
Advanced Options	
General	

Table 14. Passenger Delay Cost Calculations

Properties: All Arrived? (Decide Step Instance)	
Basic Logic	
Decide Type	**ConditionBased**
Expression	**Security.InputBuffer.NumberEntered == N**
Advanced Options	
General	

Table 15. Decision to Close Counters Logic

Properties: Close Counters (Assign Step Instance)	
Basic Logic	
State Variable Name	**Counters.CurrentCapacity**
New Value	0.0
Assignments (More)	0 Rows
Advanced Options	
General	

Table 16. Closing the Counters by Setting Capacity to Zero

b. Cost Factors

With the basic flow of the model complete, the final cost elements need to be included. The first of these is the Airline Delay Cost (see Figure 19). This cost is incurred for all passengers who are scheduled to be on the plane, but have not cleared the service counter at the time the service counter is supposed to close (Table 17). This process is called each time an entity arrives at the "Security" object. The current time is checked to determine whether or not the counters should be closed, and if they are—the Aircraft Delay Cost is invoked (see Table 18).

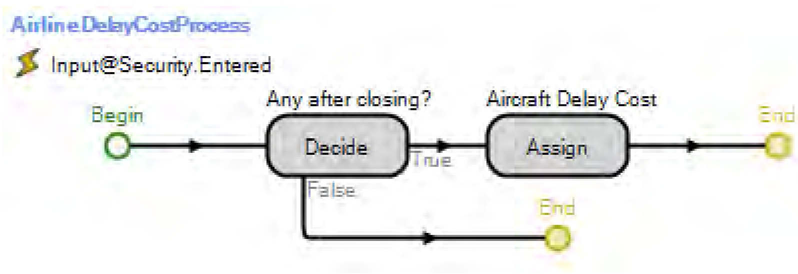


Figure 19. Airline Delay Cost Process

Properties: Any after closing? (Decide Step Instance)

Basic Logic	
Decide Type	**ConditionBased**
Expression	**TimeNow > T**
Advanced Options	
General	

Table 17. Logic to Decide Whether or Not to Implement Aircraft Delay Cost

Properties: Aircraft Delay Cost (Assign Step Instance)

Basic Logic	
State Variable Name	**TC_AirlineDelay**
New Value	**TC_AirlineDelay+h**
Assignments (More)	0 Rows
Advanced Options	
General	

Table 18. Aircraft Delay Cost Assignment

At the end of the run, the "OnRunEnding" process is called (see Figure 20). This process calculates the total cost for the run by the summation of the airline delay costs, the service counter costs, and the passenger delay costs (see Table 19). This is the key output of the simulation, on which results will be compared.

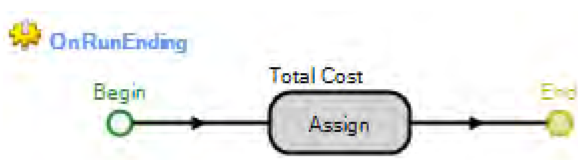


Figure 20. Total Cost Summation Process

Properties: Total Cost (Assign Step Instance)	
Basic Logic	
State Variable Name	**TC**
New Value	**TC_AirlineDelay+TC_Counters+TC_PassengerDelay**
Assignments (More)	0 Rows
Advanced Options	
General	

Table 19. Total Cost Calculation

c. Decision Management

With all necessary objects, connections, and cost calculations in place, the remaining portion of the implementation is the decision management criteria for how many service counters should be open, given the state of the system. The logic of the management process is seen in Figure 21.

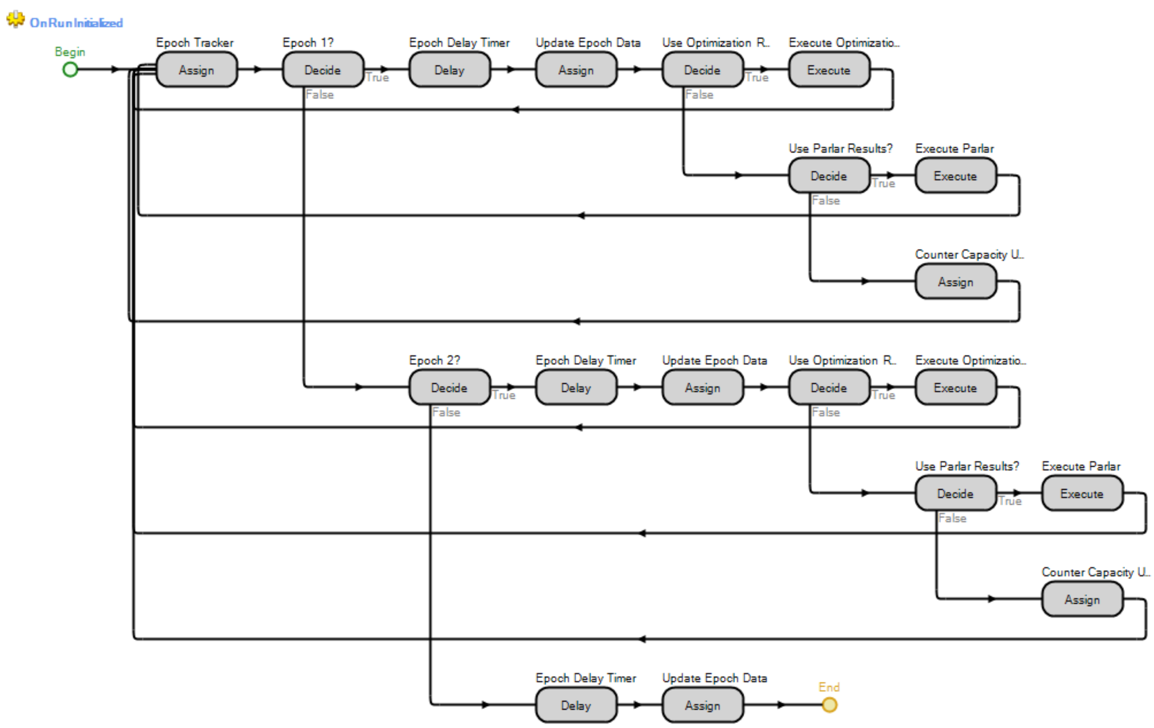


Figure 21. Management Process

Upon run initialization, this process commences its work. The first assignment is to update the variable tracking which epoch the simulation is in (see Table 20). After each epoch is complete, the process loops back to the epoch Tracker to move to the next epoch.

Properties: Epoch Tracker (Assign Step Instance)	
Basic Logic	
State Variable Name	**CurrentEpoch**
New Value	**CurrentEpoch+1**
Assignments (More)	0 Rows
Advanced Options	
General	

Table 20. Current Epoch Calculations

The next step is to determine which epoch we are in, in order to facilitate the decision-making process (see Table 21). Considering changes only occur at the end of the first and second epoch, the logic for the third epoch is much simpler, and does not have to loop back to the epoch tracker.

Properties: Epoch 1? (Decide Step Instance)	
Basic Logic	
Decide Type	**ConditionBased**
Expression	**CurrentEpoch==1**
Advanced Options	
General	

Properties: Epoch 2? (Decide Step Instance)	
Basic Logic	
Decide Type	**ConditionBased**
Expression	**CurrentEpoch==2**
Advanced Options	
General	

Table 21. Epoch Decision Logic

Figure 22 shows the logic flow for epoch 1. As the first and second epoch logic works the same, with the only difference in updating the specifics for each epoch, the discussion of the logic flow for epoch 1 applies to both.

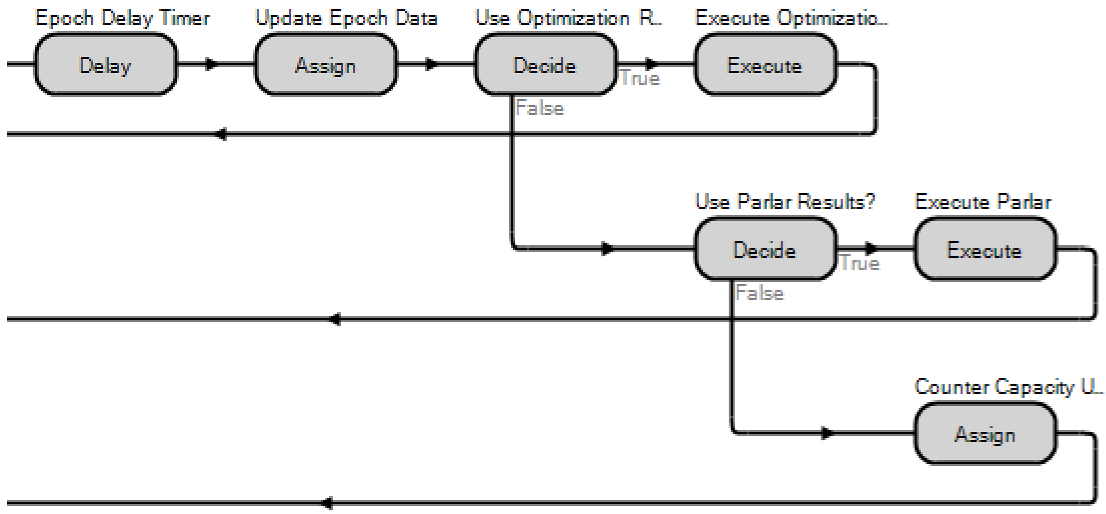


Figure 22. Logic Flow for Epochs 1 and 2

Upon reaching either epoch 1 or 2, a delay of 20 minutes occurs before proceeding to the next step. This time is based on T/K where T was the time allotted (in this case, $T=1$ hour) and the number of epochs $(K=3)$. Following the delay, the number of arrivals and the number of people served during that period are calculated in the Update epoch Data step. Simio allows for multiple variables to be manipulated with each assigned step, as shown in Figure 23.

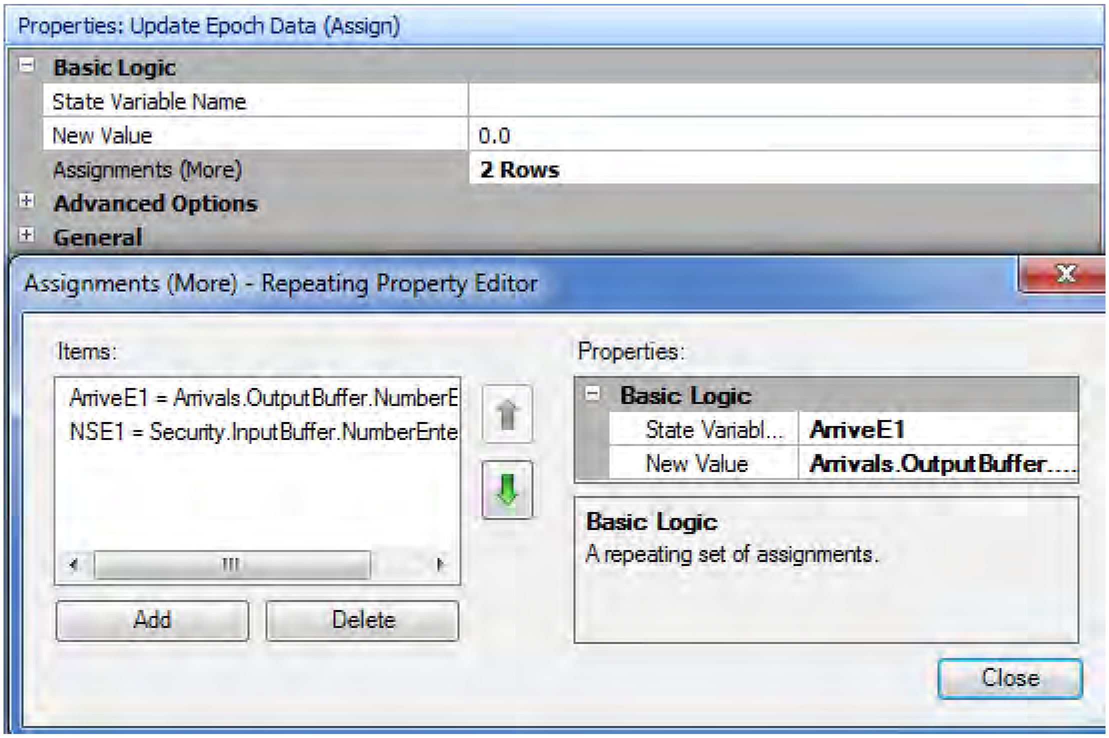


Figure 23. Updating Epoch Data

Using the "Experiment" feature of Simio, logic was included in the system to determine which type of decision-making process to use. The first option checks to see if the configuration generated by the built-in optimization process is to be used, and if it is, it executes it.

Figure 24 shows the logic flow for the optimization results. Each of the decision points within the logic is determined via the built-in optimization tool "Opt-Quest." This tool uses a heuristic algorithm and can be configured for multiple settings and variables. For each epoch, there is a decision point for the number of arrivals that have occurred from the start of the run to the end of that epoch. Based on whether or not the number of arrivals were less than or equal to that decision point, the next step of the logic is executed. The next step is based on the number of passengers served from the start of the run to the end of that epoch. Based on the number of passengers served, the "Counters" capacity is updated with a value between 1 and 5.

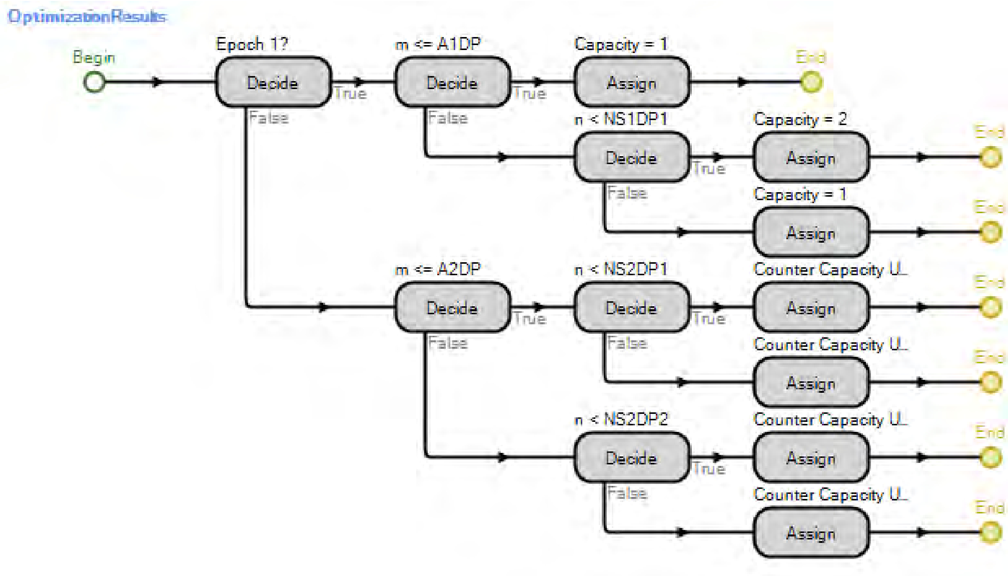


Figure 24. Optimization Results Server Capacity Assignment Logic

The results of the optimization are discussed in the analysis segment. Table 22 lists the variable names, along with their definitions and range of values. To do a full-factorial design requires a total of 100,656,875 design points. Regardless of the computing power available, running a sufficient number of replications for each of those design points could not be done in a reasonable time.

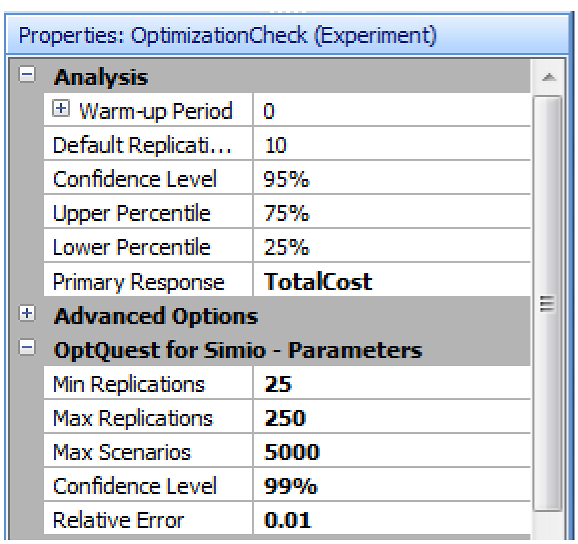
Variable Name	Description	Min Value	Max Value	Increment	Number of Settings
A1DP	*m* in epoch 1	0	10	1	11
NS1DP1	*n* in epoch 1 when "*m*" Exceeds A1DP	0	10	1	11
A2DP	*m* in epoch 2	0	10	1	11
NS2DP1	*n* in epoch 2 when "*m*" is Less than or Equal to A2DP	0	10	1	11
A2NS1_A1	Counters to open when in epoch 2 when *m* <= A2DP and *n* < NS2DP1	1	5	1	5
A2NS1_A2	Counters to open when in epoch 2 when *m* <= A2DP and *n* >= NS2DP1	1	5	1	5
NS2DP2	*n* in epoch 2 when "*m*" is Less than or Equal to A2DP	0	10	1	11
A2NS1_A3	Counters to open when in epoch 2 when *m* > A2DP and *n* < NS2DP2	1	5	1	5
A2NS1_A4	Counters to open when in epoch 2 when *m* > A2DP and *n* >= NS2DP1	1	5	1	5
Number of combinations required for a full-factorial design:					100,656,875

Table 22. Variable Names and Descriptions for Optimization Run

In order to reduce the number of possible combinations, insight from the previous design runs was utilized to limit the scope of each of the variables. The updated values are seen in Table 23. It was with these values that the initial optimization run was conducted. The initial configuration for the optimization run allowed for a minimum of 25 replications, a maximum of 250 replications, and a total of 5,000 scenarios—with the confidence level being 99% and a relative error of 0.01 (see Table 24).

Variable Name	**Description**	**Min Value**	**Max Value**	**Increment**	**Number of Settings**
A1DP	*m* in epoch 1	2	6	1	5
NS1DP1	*n* in epoch 1 when "*m*" Exceeds A1DP	3	6	1	4
A2DP	*m* in epoch 2	5	8	1	4
NS2DP1	*n* in epoch 2 when "*m*" is Less than or Equal to A2DP	3	6	1	4
A2NS1_A1	Counters to open when in epoch 2 when *m* <= A2DP and *n* < NS2DP1	2	8	1	7
A2NS1_A2	Counters to open when in epoch 2 when *m* <= A2DP and *n* >= NS2DP1	2	8	1	7
NS2DP2	*n* in epoch 2 when "*m*" is Less than or Equal to A2DP	2	8	1	7
A2NS1_A3	Counters to open when in epoch 2 when *m* > A2DP and *n* < NS2DP2	2	4	1	3
A2NS1_A4	Counters to open when in epoch 2 when *m* > A2DP and *n* >= NS2DP1	2	4	1	3
Number of combinations required for a full-factorial design:					987,840

Table 23. Changed Variable Min and Max Values to Utilize for Optimization Run


Properties: OptimizationCheck (Experiment)

Analysis	
Warm-up Period	0
Default Replicati...	10
Confidence Level	95%
Upper Percentile	75%
Lower Percentile	25%
Primary Response	**TotalCost**
Advanced Options	
OptQuest for Simio - Parameters	
Min Replications	**25**
Max Replications	**250**
Max Scenarios	**5000**
Confidence Level	**99%**
Relative Error	**0.01**

Table 24. Configuration for First Optimization Run

Examining the top 100 scenarios for the minimum and maximum values of each variable, the optimization run was conducted again, this time requiring a minimum of 100 and a maximum of 500 replications. This process was completed once more for the top eight scenarios, but now with a minimum of 2,000 and a maximum of 4,000 replications, with the confidence level being 99.9% and a relative error of 0.001 (see Table 25).

Properties: OptimizationCheck (Experiment)

Analysis	
Warm-up Period	0
Default Replicati...	10
Confidence Level	95%
Upper Percentile	75%
Lower Percentile	25%
Primary Response	**TotalCost**
Advanced Options	
OptQuest for Simio - Parameters	
Min Replications	**2000**
Max Replications	**4000**
Max Scenarios	**5000**
Confidence Level	**99.9%**
Relative Error	**0.001**

Table 25. Configuration for Final Optimization Run

After conducting a total of 4,000 runs on the each of the eight scenarios, a subset of 29,528 were examined (based on 10 arrivals during the 1-hour period). The summary statistics of these are listed in Table 26. Utilizing a Student's t comparison method, all scenarios are statistically the same. As scenario 4 had the lowest mean and was in the group with the smallest range, scenario 4's configuration (see Table 27) is determined as the final optimized solution parameters.

Scenario	Mean (TotalCost)	Std Dev (TotalCost)	Min (TotalCost)	Max (TotalCost)
1	165.164	48.144	67.781	435.174
2	165.128	47.373	71.784	435.174
3	164.998	47.519	71.784	435.174
4	164.961	47.535	71.784	435.174
5	166.476	49.259	67.781	477.470
6	166.256	49.019	67.781	477.470
7	165.140	47.427	71.784	435.174
8	165.263	47.988	67.781	435.174

Table 26. Summary Statistics for Eight Optimization Candidates

Variable Name	Description	Final Value
A1DP	*m* in epoch 1	3
NS1DP1	*n* in epoch 1 when "*m*" Exceeds A1DP	2
A2DP	*m* in epoch 2	8
NS2DP1	*n* in epoch 2 when "*m*" is Less than or Equal to A2DP	4
A2NS1_A1	Counters to open when in epoch 2 when *m* <= A2DP and *n* < NS2DP1	3
A2NS1_A2	Counters to open when in epoch 2 when *m* <= A2DP and *n* >= NS2DP1	2
NS2DP2	*n* in epoch 2 when "*m*" is Less than or Equal to A2DP	6
A2NS1_A3	Counters to open when in epoch 2 when *m* > A2DP and *n* < NS2DP2	3
A2NS1_A4	Counters to open when in epoch 2 when *m* > A2DP and *n* >= NS2DP1	2

Table 27. Final Variable Settings for Optimization Run Configuration

If the results from the optimization analysis are not to be used, the logic then checks to see if the Parlar solution is to be used. The implementation of the Parlar solution follows a similar, but more complex logic tree, as the optimization results uses. If neither complex management solution is to be used, the system defaults to a preset value for the number of service counters to remain open during an epoch. At the completion of epochs 1 and 2, the process is looped back to the beginning to increment the epoch counter and follow the logic flow again. At the end of the third epoch, the process stops after recording the number of arrivals to the system and the number who have completed service.

3. Design of Experiments and Analysis

The objective of this project is to determine if the Parlar results provide an optimal solution when compared to a generic baseline configuration, a configuration based on insight provided from the simulation results, and a solution gained from using Simio's built-in optimization algorithm. In addition to testing the optimality of the Parlar Model, two statements regarding the optimality of the Parlar Model's solution will be assessed:

- The number of counters to keep open is nondecreasing in the number of passengers who have already arrived (monotonicity of the optimal solution).
- The number of counters to open is nonincreasing in the number of passengers who have been serviced. (Parlar & Sharafali, 2008)

To establish the generic baseline configuration, a full-factorial design was utilized, allowing the number of servers to be open in each epoch range from 1 to 5, with each design point utilizing 100 replications. Using the resulting output, and filtering out instances when 10 arrivals did not occur within the 1-hour time period, 11,422 simulation results remained. Utilizing a Student's t comparison method, the top 30 results with the lowest mean cost are selected for further experimentation with an increased number of runs. These top 30 were then selected for an additional 400 runs, with the comparison made again, now selecting the top five configurations. These final five configurations were then subjected to a total of 4,000 replications each, with the configurations for each scenario shown in Table 28.

Configuration	Epoch 1 Capacity	Epoch 2 Capacity	Epoch 3 Capacity
Scenario2	1	1	2
Scenario3	1	1	3
Scenario7	1	2	2
Scenario8	1	2	3
Scenario4	1	1	4

Table 28. Scenario Configuration Matrix

For each of the scenarios, the values listed in the applicable epoch Capacities column represent the number of service counters open during that period. For example, Scenario 2 has one server open during epoch 1, one server open during epoch 2, and two servers open during epoch 3. Figure 25 shows the distribution of cost values for each of the scenarios. Utilizing a Student's t comparison method, as discussed previously, Scenario 2 had the lowest mean and was significantly different than the other scenarios (see Table 29).

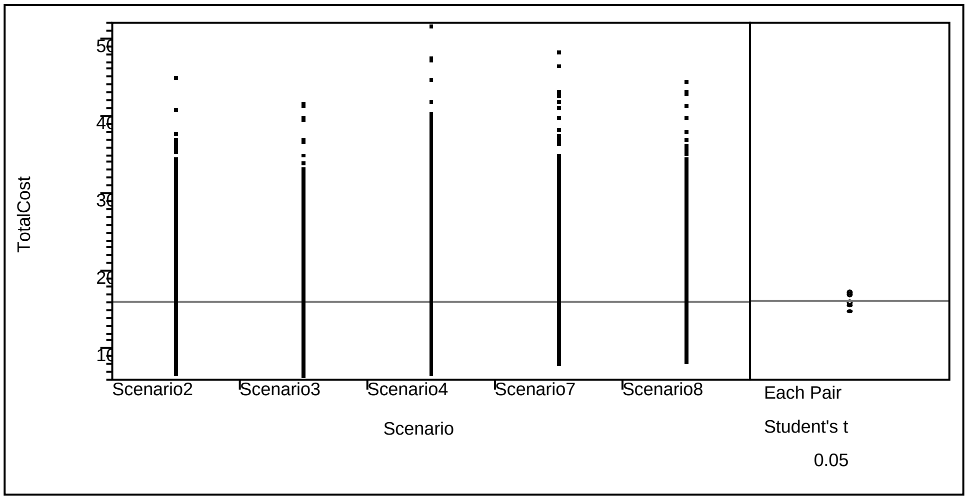


Figure 25. One-Way Analysis of Cost by Scenario

Level		Mean
Scenario4	A	172.44781
Scenario8	B	169.58419
Scenario7	C	159.83866
Scenario3	D	156.30028
Scenario2	E	148.36898

Levels not connected by same letter are significantly different.

Table 29. Means Comparisons for Each Pair Using Student's t

With the generic baseline configuration established, one can exploit the benefits of simulation to analyze the underlying conditions that cause the variability within the cost data, as using simulation lets you examine the conditions which cause the outliers—if the appropriate parameters are recorded during the simulation run. After conducting a total of 20,000 runs on the baseline configuration, a subset of 18,472 were examined (based on 10 arrivals during the 1-hour period). The summary statistics of these are listed in Table 30.

COST			
Mean	**Std Dev**	**Min**	**Max**
$148.08	$45.97	$57.49	$553.07

Table 30. Summary Analysis of 18,472 Baseline Runs

As the intent is to minimize cost, the instances in which the cost of the run exceeds the mean are examined. Through selecting a subset of data in which the total

cost was three or more standard deviations above the mean (see Table 31), we can examine the conditions under which these outliers occur.

	Number of Rows	Total Cost	Passenger Cost	Counter Cost	Arrivals Epoch 1	Arrivals Epoch 2	Served Epoch 1	Served Epoch 2
>=3 Standard Deviations	248	$319.85	$89.54	$133.53	1.33	4.58	0.53	1.91
< 3 Standard Deviations	18,224	$145.74	$57.30	$76.40	1.81	6.67	0.79	3.45

Table 31. Summary Analysis of Instances with Total Cost Greater Than or Equal to Three Standard Deviations Compared to Those With Less Than Three Standard Deviations

With the counters closing on the completion of service to the last passenger, the Counter Cost component to the total cost is significant when passengers are arriving later, forcing the counters to stay open longer. This is seen in Table 31, with the increased counter cost, and lower number of passengers served during epoch 2 (Served epoch 2) than the corresponding values in the case with less than three standard deviations in total cost.

Table 32 allows for us to identify two issues. First, counter cost is greater when the counters have to remain open for a longer period due to late arrivals. When limited to changing the number of open counters only at the designated intervals, the only option to minimize this cost is to have fewer counters open during epoch 3. This modification, however, brings the risk of increasing the passenger delay cost. To investigate this, a modification was included that set the number of open counters in epoch 3 to one when the number served in epoch 2 equaled five. Executing an additional 20,000 runs with this modification yielded the results of Tables 33 and 34. As expected, the counter operating costs decreased; however, the passenger delay costs increased too much, resulting in a higher overall mean cost.

	Mean Values							
Counter Dominant	**Number of Rows**	**Total Cost**	**Passenger Cost**	**Counter Cost**	**Arrivals Epoch 1**	**Arrivals Epoch 2**	**Served Epoch 1**	**Served Epoch 2**
Counter	191	$315.48	$79.69	$143.01	1.29	4.53	0.60	2.17
Pax	57	$334.52	$122.57	$101.78	1.46	4.75	0.28	1.02

Table 32. Summary Analysis of Instances With Total Cost Greater Than or Equal to Three Standard Deviations Based on Major Cost Component

	Mean Values							
Counter Dominant	**Number of Rows**	**Total Cost**	**Passenger Cost**	**Counter Cost**	**Arrivals Epoch 1**	**Arrivals Epoch 2**	**Served Epoch 1**	**Served Epoch 2**
Counter	31	$366.70	$92.41	$151.06	0.65	3.32	0.29	1.26
Pax	291	$388.36	$170.13	$90.12	2.11	6.56	1.24	3.43

Table 33. Summary Analysis of Instances (Using Modified Baseline) With Total Cost Greater Than or Equal to Three Standard Deviations Based on Major Cost Component

COST			
Mean	**Std Dev**	**Min**	**Max**
$158.25	$59.10	$58.66	$655.97

Table 34. Summary Analysis of 18,476 Modified Baseline Runs

Leveraging the ability to assign a variable and run multiple experiments of the simulation, a variable X was inserted into the logic to see if any value of number served in epoch 2 would enable the closing of one counter in epoch 3 (see Figure 26).

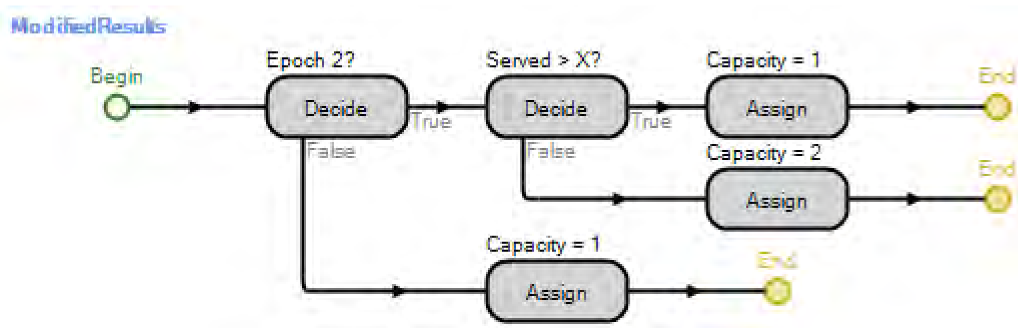


Figure 26. Branching Condition for Modified Baseline—Using Variable X, Controlled in the Experiments Window to Determine Branching

Executing 5,000 runs at each configuration, the top three results—compared to the baseline simulation—are seen in Table 35. The only scenario that was able to achieve a lower mean cost than the baseline was to only allow one counter to be open during epoch 3, when the number of served passengers during epoch 2 was greater than 7. While this does show an improvement, when a means comparison is done using Student's t test, none of the scenarios is statistically significantly different from the other (see Table 36).

COST				
Scenario	**Mean**	**Std Dev**	**Min**	**Max**
Baseline	148.66	46.59	63.71	449.56
X = 6	148.47	46.98	66.37	449.56
X = 7	148.32	46.74	66.37	449.56
X = 8	148.37	46.71	63.71	449.56

Table 35. Comparison of Generic and Modified Baseline Statistics

Level	- Level	Difference	Std Err Dif	Lower CL	Upper CL	p-Value
Baseline	X = 7	0.3379828	0.9729054	−1.56900	2.244967	0.7283
Baseline	X = 8	0.2914047	0.9729054	−1.61558	2.198389	0.7645
Baseline	X = 6	0.1887487	0.9729054	−1.71824	2.095733	0.8462
X = 6	X = 7	0.1492341	0.9729054	−1.75775	2.056219	0.8781
X = 6	X = 8	0.1026560	0.9729054	−1.80433	2.009641	0.9160
X = 8	X = 7	0.0465781	0.9729054	−1.86041	1.953563	0.9618

Table 36. Comparisons for Each Pair Using Student's t Test

The other insight gained from the original data is that when there are few arrivals in epochs 1 and 2, and few served by the end of epoch 2, the mean cost is higher. Using the same concept as before, a variable X was assigned to the modified baseline (see Figure 27) and the experiment was run 5,000 times at each design point $X \in \{0,1,2,3,4\}$ to determine if any improvement could be made. In order to see the results clearly, the conditional statement used is (Arrive < X && Served <= 0).

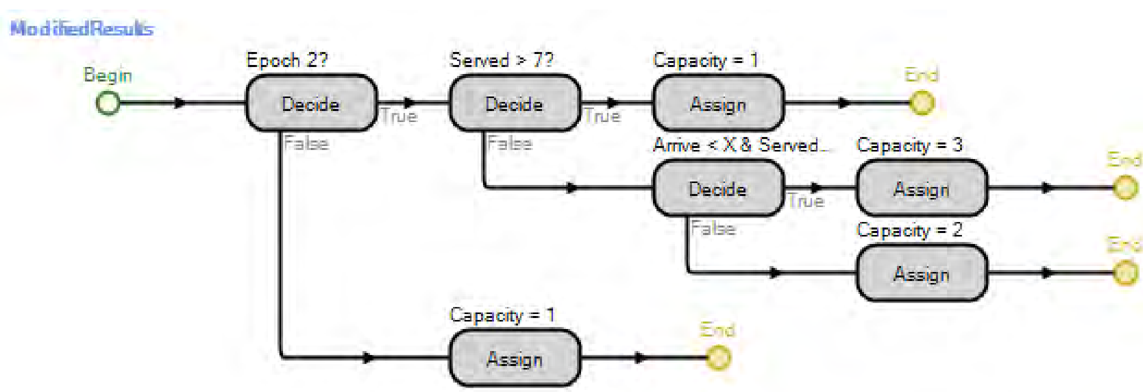


Figure 27. Branching Condition for 2nd Modified Baseline—Using Variable X, Controlled in the Experiments Window to Determine Branching

Of the 30,000 combined runs (5,000 per simulation), 27,713 runs met the previously stated criteria of all $N = 10$ arrivals occurring during the $T = 1$-hour time frame. The summary statistics for these runs are listed in Table 37.

COST				
Scenario	Mean	Std Dev	Min	Max
Mod 7 w/o Change	148.32	46.74	66.37	449.56
Mod 7 X=0	148.32	46.74	66.37	449.56
Mod 7 X=1	148.33	46.76	66.37	449.56
Mod 7 X=2	148.33	46.72	66.37	449.56
Mod 7 X=3	148.30	46.70	66.37	449.56
Mod 7 X=4	148.36	46.56	66.37	447.46

Table 37. Comparison of Modified Baseline (More Than 7 Served in Epoch 2), With Opening an Additional Counter in Epoch 3 Based on Number of Arrivals (With None Completing Service by the End of Epoch 2)

Utilizing the result with the lowest mean cost (when the number of arrivals in epoch 2 is less than 3), one additional set of experiments is conducted. Given the conditional statement (Arrive < 3 && Served <= X), where $X \in \{0,1,2,3\}$, we conducted 10,000 replications at each design point. Based on the results listed in Table 38, there was an improvement when the X value was set to two and three. Considering that the results in both cases are identical, the finalized modification has the conditional statement set at (Arrive < 3 && Served <= 2). The finalized implementation is seen in Figure 28.

COST				
Scenario	Mean	Std Dev	Min	Max
Mod S7 A3 X=0	148.30	46.70	66.37	449.56
Mod S7 A3 X=1	148.21	46.58	66.37	449.56
Mod S7 A3 X=2	148.13	46.37	66.37	449.56
Mod S7 A3 X=3	148.13	46.37	66.37	449.56

Table 38. Comparison of Modified Baseline (More Than 7 Served in Epoch 2), With Opening an Additional Counter in Epoch 3 Based on 3 Arrivals (and Less Than or Equal to X Completing Service by the End of Epoch 2)

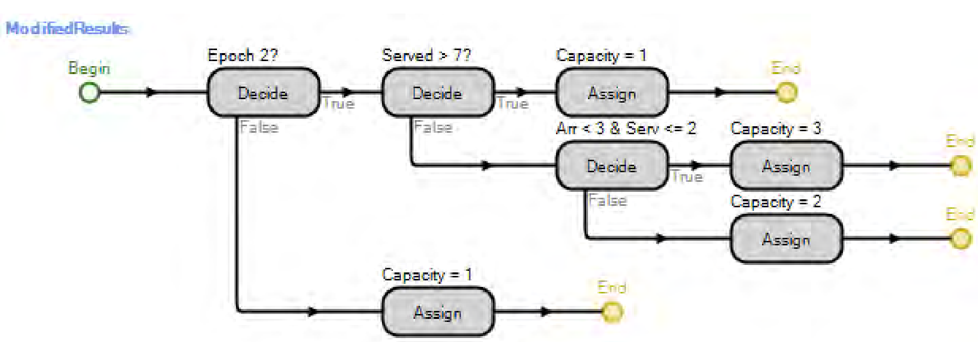


Figure 28. Finalized Modified Baseline Configuration

With the modified baseline configuration determined, the final comparison between the four models can now be assessed. Utilizing a total of 80,000 runs (20,000 of each configuration), 73,879 runs met the 10 arrival requirement. The summary statistics of the four different configurations is listed in Table 39. Further discussion of these results is in the next section.

COST				
Scenario	**Mean**	**Std Dev**	**Min**	**Max**
Baseline	148.08	45.97	57.49	553.07
Modified Baseline	147.59	45.73	57.49	449.56
Optimization Results	151.71	47.44	57.49	467.35
Parlar Model	154.28	48.78	57.49	510.45

Table 39. Comparison of Four Scenarios

With the results having very similar values, a one-way analysis was conducted that showed that the Modified Baseline and Baseline were statistically comparable; however, the Parlar Model and the Optimization Results configurations were not (see Tables 40 and 41), and had higher average costs.

Level				Mean
Parlar Model	A			154.27976
Optimization Results		B		151.71097
Baseline			C	148.07510
Modified Baseline			C	147.59466

Table 40. Means Comparisons for Each Pair Using Student's t Test

Level	- Level	Difference	Std Err Dif	Lower CL	Upper CL	p-Value
Parlar Model	Modified Baseline	6.685103	0.4890703	5.07574	8.294466	<.0001*
Parlar Model	Baseline	6.204661	0.4890306	4.59543	7.813894	<.0001*
Optimization Results	Modified Baseline	4.116312	0.4890901	2.50688	5.725741	<.0001*
Optimization Results	Baseline	3.635871	0.4890504	2.02657	5.245169	<.0001*
Parlar Model	Optimization Results	2.568791	0.4890504	0.95949	4.178088	<.0001*
Baseline	Modified Baseline	0.480441	0.4890703	–1.12892	2.089805	0.3259

Table 41. Means Comparison of Each Configuration With Corresponding p-Values

4. Document and Brief Results

As discussed previously, it is essential that the insight learned within a simulation study is conveyed to the decision maker in a manner that is clear and concise, addressing the questions raised by the study. From the analysis of the four different model

configurations, the following are the significant findings (with the optimal resulting design output, as previously shown in Figure 28):

- The configuration leading to the lowest mean cost was the Modified baseline.
- Contrary to the statement of monotonicity by Parlar and Sharafali, monotonicity does not exist in the optimal solution.
- The finding by Parlar and Sharafali that the number of counters to open is nonincreasing in the number of passengers who have been serviced held true.
- While the Modified baseline configuration had a lower mean than any other configuration, it did not have a statistically significant difference with that of the baseline configuration.

D. USING SIMULATION TO EXPAND OUR UNDERSTANDING

With the comparison between the analytical model of Parlar and Sharafali and a simulation complete, let us now turn to demonstrating the way simulation can expand our understanding of a situation. Utilizing the same arrival profiles, number of confirmed passengers, associated costs, and number of available counters, we shall examine the two following changes to the simulation:

- Service rate does not increase as a function of the number of personnel in line.
- The service rate is approximated using a gamma distribution, with parameters $(\alpha = 6, \beta = 2)$.

1. Service Rate Does Not Change

Utilizing the baseline configuration previously discussed, 8,000 simulation runs were conducted, with 4,000 of those runs utilizing the increased service rate (as discussed previously), and the other 4,000 runs utilizing a nonadjusting service rate with the same mean time of service as before. It took less than two minutes to make the changes in the scenario, conduct the runs, and view the resulting statistics. As expected, the mean cost for the configuration without the increased service rate is higher (see Table 42).

	Mean Values			
Scenario	Total Cost	Passenger Cost	Counter Cost	Service Rate
Constant Service Rate	388.41	147.81	147.67	12.0000
Increased Service Rate	148.37	57.46	77.14	4.0701

Table 42. Constant Versus Increased Service Rate Comparison of Mean Values

With these results being assessed so quickly, as conducted for the initial study, a full-factorial design was utilized to test all possible combinations of counters being open in each epoch. Running these 125 design points, with 1,000 repetitions each, took a total of three minutes, the majority of which was utilized in setting up the optimization interface. Taking the top 10 design points and utilizing a Student's t test to compare means, the top five scenarios had no statistically significant difference. Conducting a further 4,000 runs per design point (for a total of 5,000 each). The total elapsed time to conduct all the experiments was just under nine minutes. The optimal baseline configuration when the service rate is not increased as a function of the number of people who have arrived, but not been served, is one counter in epoch 1, three counters in epoch 2, and three counters in epoch 3. The results of these runs are listed in Table 43.

	Mean Values			
Scenario	Total Cost	Passenger Cost	Counter Cost	Airline Delay Cost
Servers: 1 3 3	319.04	79.62	190.76	48.66
Servers: 1 2 3	322.85	90.52	177.19	55.15
Servers: 1 4 3	326.37	73.79	207.58	44.99
Servers: 1 2 4	326.87	79.70	203.36	43.81
Servers: 1 3 2	329.57	97.92	167.84	63.82

Table 43. Optimal Baseline Counter Schedule for System When Service Rate is Not Increased as a Function of the Number of Passengers in the System Who Have Not Completed Service

2. Service Rate Uses Gamma Distribution

Continuing from the previous example, the same problem is examined utilizing a service rate based on a gamma distribution. As the intention is to maintain the same mean service time, the parameters utilized for the gamma distribution should provide the same mean service time as that of the exponential service rate. As the previous scenario utilized an exponential distribution with a mean number passengers having service per hour $\mu = 5$, the expected service time was 12 minutes. Corresponding parameters for the

gamma distribution $(\alpha = 6, \beta = 2)$ achieve the same expected service time. Figure 29 illustrates the difference in the two distributions.

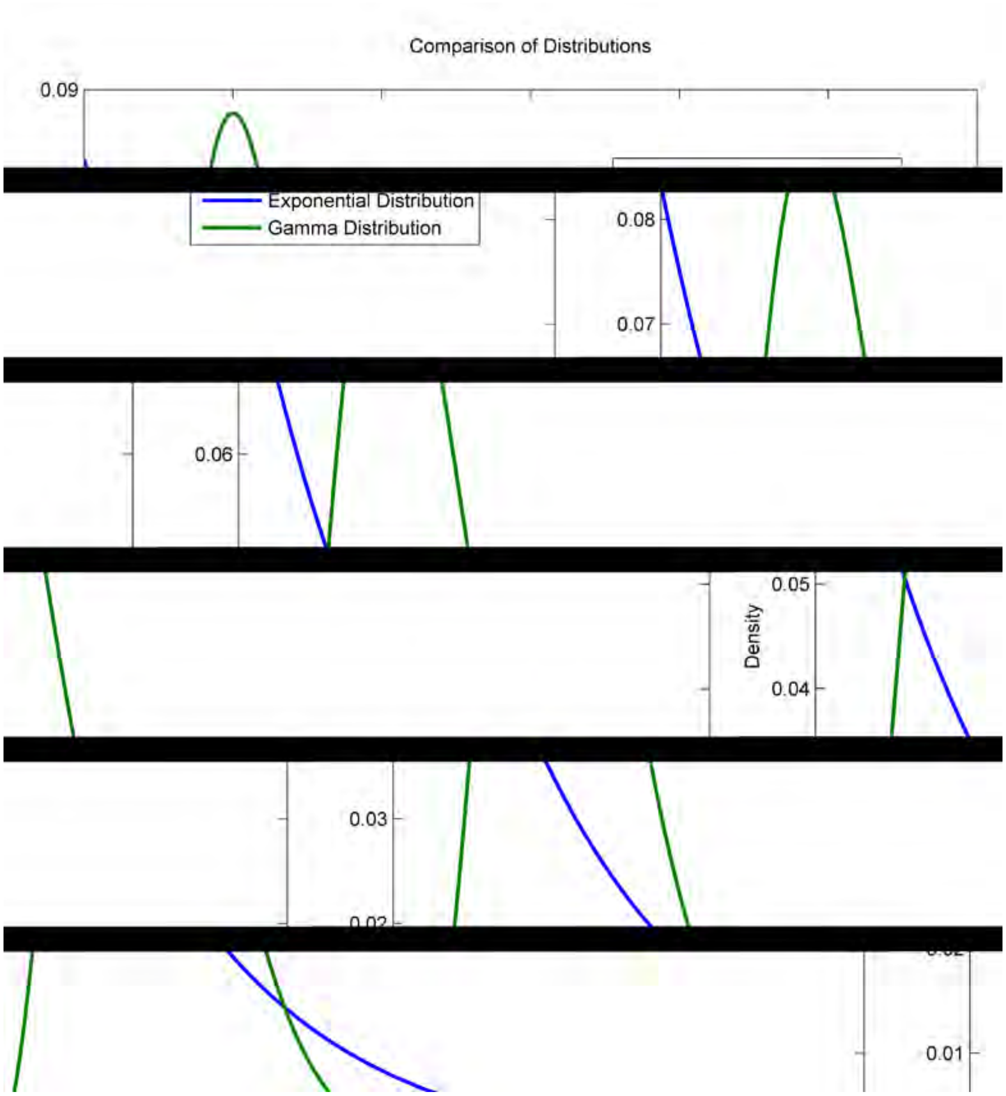


Figure 29. Comparison of Exponential to Gamma Distributions With the Same Mean Times

With the two distributions presented visually, one must question the validity of the use of an exponential service time in the first place. In the case of service times, when looking up the "Random.Exponential(mean)" function in the *Simio Reference Guide*, the following statement appears: "This distribution is generally not appropriate for modeling process delay times" (Simio, 2011). The effect of choosing the wrong

distribution for generating service times can be substantial. In this case, in an analysis of 10,000 simulation runs (5,000 for each distribution), the mean cost of using a gamma distribution was lower than that of the exponential distribution of the same mean (see Table 44). The fact that the mean is lower is not the important insight gained from the comparison; rather, the true insight lies within the distribution of the total costs. Examining Figures 30 and 31 clearly shows that the use of the exponential service times is producing a much broader distribution of costs than that of the gamma distribution. With a much narrower distribution of costs using the gamma distribution, the "optimal policy" based on an exponential service time may no longer be valid.

Scenario	Mean Values						
	Cost Data				Passenger Data		
	Total Cost	Passenger Cost	Counter Cost	Airline Delay Cost	Time In System	Minimum Time	Maximum Time
Exponential	319.04	79.62	190.76	48.66	15.400	2.710	38.241
Gamma	302.32	89.25	166.64	46.43	15.742	7.328	26.220

Table 44. Comparison of Mean Values When Service Time is Assumed Exponential to Service Time Assumed Gamma

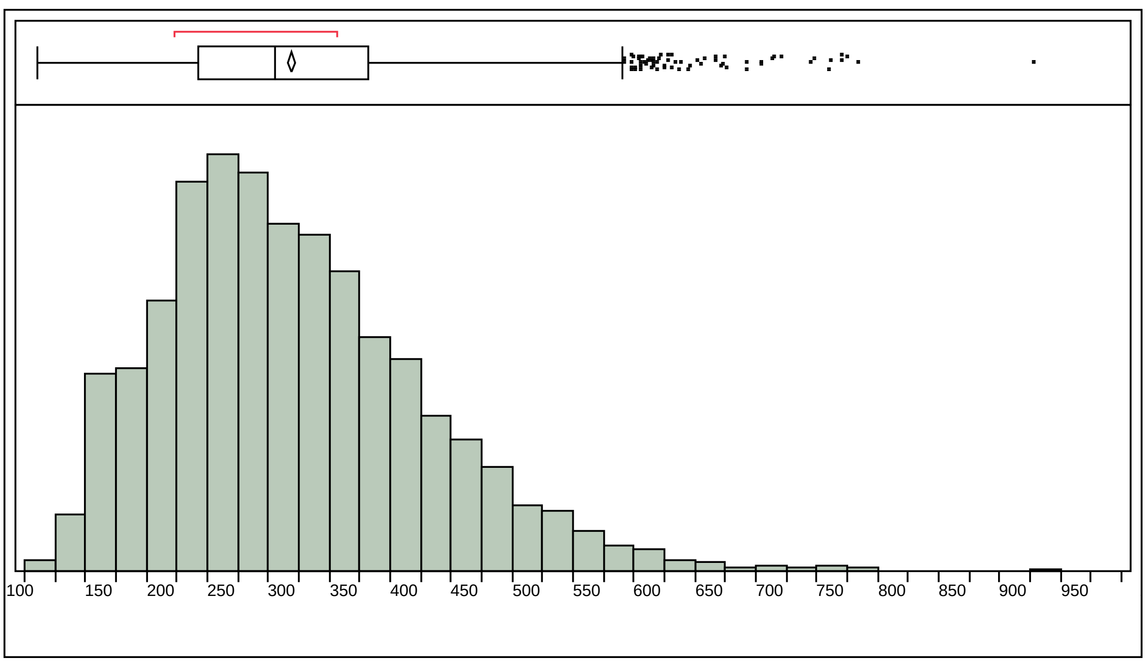


Figure 30. Distribution of Total Costs When Exponential Service Time is Assumed

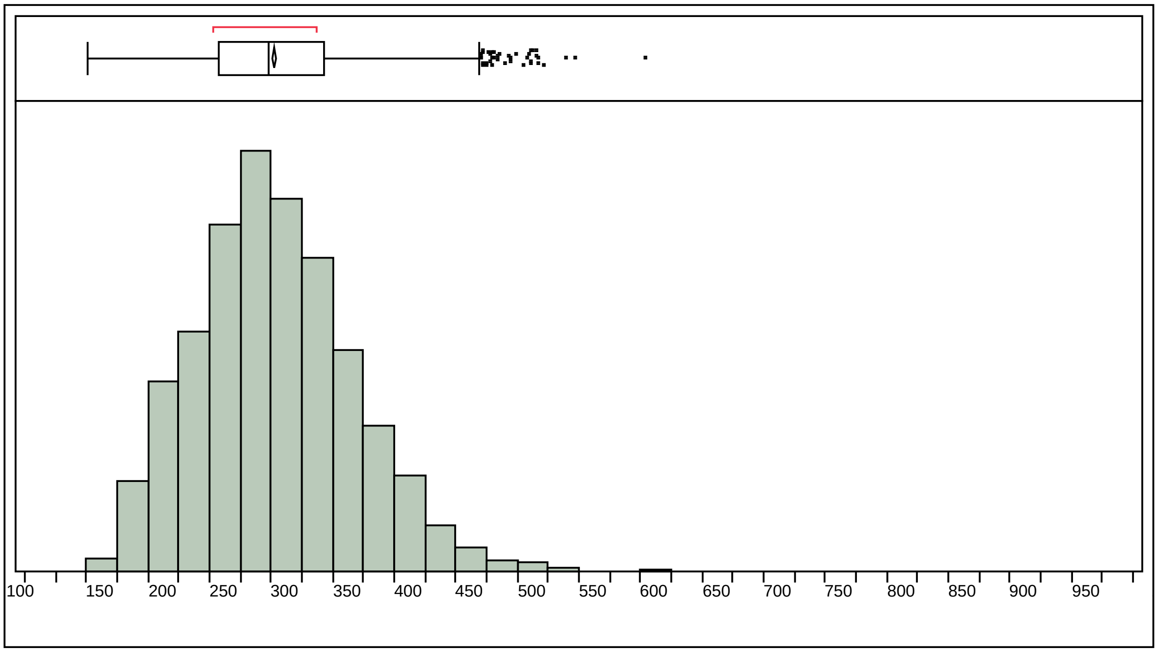


Figure 31. Distribution of Total Costs When Gamma Service Time is Assumed

IV. CONCLUSIONS

A. WHY SIMULATION SHOULD BE A METHOD OF FIRST RESORT

The days of limited access to computers, simulation software, and modeling experts has long passed, yet the operations research community is still plagued by the adage that simulation is a "method of last resort." In an attempt to change the paradigm, this thesis has demonstrated how simulation has changed since its inception. With a brief overview of the history of simulation, to identifying the increase in computing power, insight was given to the reader on where simulation started and where it is today.

With an understanding of the changes in simulation over the decades, this thesis then demonstrated the power of simulation when faced with a complex problem over that of an analytical solution. Often, to make a complex solution tractable, many assumptions have to be made, which are often in conflict with how the system really works. With today's robust simulation software capabilities, many of these assumptions (e.g., normality, independence, memorylessness, deterministic, linear, stationary, and homoscedasticity) are not required to be made. In a detailed examination of Parlar and Sharafali's (2008) paper "Dynamic Allocation of Airline Check-In Counters: A Queueing Optimization Approach," this thesis identified the vast number of assumptions that had to be made in order to make this solution analytically tractable. Then, through the use of a simulation study utilizing as many of the assumptions and conditions in the Parlar Model as possible, a comparison was made between the simulation and analytical results.

When the restrictions of the Parlar Model were removed from the simulation, this thesis demonstrated how simulations can easily adapt to changes, often providing much more insight on the workings of a system, as well as enabling analysts to find not only the optimal design, but robust designs as well.

B. SUMMARY OF COMPARISON BETWEEN THE SIMULATION AND ANALYTICAL RESULTS OF THE AIRLINE COUNTER QUEUE

This thesis provides an examination of a paper written by Parlar and Sharafali (2008) on an analytical approach to providing an optimal solution on the allocation of airline check-in counters. Numerous assumptions were made by Parlar and Sharafali in order to present an analytically tractable problem. Many of these assumptions are made contrary to the actual workings of the system and, at times, were completely illogical.

With an understanding of the model presented by Parlar and Sharafali, this thesis then presents the methodology originally presented by Law and Kelton (1982), and adopted by the Defense MSCO for VV&A (MSCO, 2001). Through this process, and the use of Simio simulation software, this thesis produced a simulation model, copying the assumptions and distributions of the Parlar and Sharafali model as closely as possible. The simulation model was then exercised to test the validity of Parlar and Sharafali's findings of the optimal configurations, monotonicty within the optimal solution, and that the number of counters to open is nonincreasing in the number of passengers who have been serviced.

With the simulation model mimicking the Parlar and Sharafali's model as closely as possible, the results indicated that Parlar and Sharafali's solution was not optimal, and that monotonicity did not exist in the solution. After disproving these two points, the simulation demonstrated the effect two of the assumptions had on their analysis. First, Parlar and Sharafali assumed that service rate would increase as a multiple of the number of people in line. This assumption played a dramatic role in the development of the optimal solution, which if attempted in reality, would be ineffective. The second demonstration illustrated the importance of choosing the right distribution model. Parlar and Sharafali used an exponential service time, which is also not representative of reality. Utilizing a gamma distribution with the same mean value demonstrated that the average total cost of a run is lower, and from a much narrower range, than when calculated with the exponential distribution due to the heavier weightings of the exponential distribution at higher numbers.

C. RECOMMENDATIONS FOR FUTURE RESEARCH

More literature could exist on the topics of when to utilize simulation, how to choose the correct simulation software, and the importance of using advanced DOE techniques when conducting a simulation study. Future research relating to the use of simulation can extend to the following:

- Guidance on how to choose a simulation software suite based on the nature of the problem to be studied.
- Analysis of simulation study results, which utilized an inadequate DOE.
- The effects of incorrect assumptions of input data (e.g., normality, independence, memorylessness, deterministic, linear, stationary, and homoscedasticity) on simulation study results when input assumptions were made for convenience.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Ancker, C. J. (1995). A proposed foundation for a theory of combat. *Naval Research Logistics*, *42*, 311.

Army Modeling and Simulation Office (AMSO). (2011). History of modeling & simulation. Retrieved from http://www.ms.army.mil/about/history/index.html

Balci, O. (1997). Verification, validation and accreditation of simulation models. *Proceeedings of the 1997 Winter Simulation Conference*, Atlanta, GA, 135–141.

Balci O., & Sargent, R. G. (1981). A methodology for validating multivariate response simulation models by using simultaneous confidence intervals. (Technical Report CS81019-R). Blacksburg, VA: Virginia Polytechnic Institute

Basawa, I. V., & Prakasa Rao, B. L. S. (1980). *Statistical inference for stochastic processes*. New York: Academic Press.

Bhatia, P. (2010). MCA-504 systems simulation and modeling class notes. Unpublished manuscript. Retrieved from http://210.212.119.186/studymaterial/mca-5/mca-504.pdf

Bohanner, D. (2004). Simulations prepare Marine Corps for war. Retrieved from http://www.defense.gov/transformation/articles/2004-12/ta120204d.html

Cioppa, T. M., & Lucas, T. W. (2007). Efficient nearly orthogonal and space-filling Latin hypercubes. *Technometrics*, *49*(1), 45.

Committee on Modeling and Simulation for Defense Transformation (2006). *Defense modeling simulation, and analysis – Meeting the challenge (1st ed.).* Washington, D.C.: National Academies Press.

Cremeans, J. E. (1967). Why simulation? (Paper No. RAC-P-30). McLean, VA.

Dongarra, J., Meuer, H., Simon, H. D., & Strohmaier, E. (2000). High performance computing today. No. LBNL Paper LBNL-45666).Lawrence Berkeley National Laboratory. Retrieved from http://escholarship.org/uc/item/81m1d8cx

Fisher, R. A. (1925). Applications of 'student's' distribution. *Metron*, *5*, 90–104.

Fritz, W. B. (1994). ENIAC-A problem solver. *IEEE Annals of the History of Computing*, *16*(1), 25–45.

Goldsman, D., Nance, R. E., & Wilson, J. R. (2009). A brief history of simulation. *Proceeedings of the 2009 Winter Simulation Conference*, Austin, TX, 310–313.

Hammersley, J. M., & Handscomb, D. C. (1964). *Monte Carlo methods*. London & New York: Chapman and Hall.

Harling, J. (1958). Simulation techniques in operational research. *OR*, *9*(1), 9–21.

Hillier F. S., & Lieberman, G. J. (1986) *Introduction to operations research* (4th ed). Oakland, CA: Holden-Day.

Joines, J. A., & Roberts, S. D. (2010). *Simulation modeling with simio: A workbook*. Raleigh, NC: Simio, LLC.

Keller, L., Harrell, C., & Leavy, J. (1991). The three best reasons why simulation fails. *Industrial Engineering*, *23*(4), 27–31.

Kelton, W. D. (2000). Experimental design for simulation. *Proceedings of the 2000 Winter Simulation Conference*, Orlando, FL, *1*, 32–38.

Kimball, A. W. (1957). Errors of the third kind in statistical consulting, *52*(278), 133–142.

Kiviat, P. J. (1967). Digital computer simulation: Modeling concepts. (MEMORANDUM No. RM-5378-PR). Santa Monica, CA: The RAND Corporation.

Kleijnen, J. P. C., Sanchez, S. M., Lucas, T. W., & Cioppa, T. M. (2005). A user's guide to the brave new world of designing simulation experiments. *INFORMS Journal on Computing*, *17*(3), 263–289.

Komorowski, M. (2009). A history of storage costs. Retrieved from http://www.mkomo.com/cost-per-gigabyte

Laplace, P. S. (1812). In Courcier (Ed.), *Théorie analytique des probabilités*. Paris, France.

Law, A. M. (2007). *Simulation modeling and analysis* (4th ed.). New York, NY: McGraw-Hill Higher Education.

Law, A. M., & Kelton, W. D. (1982). *Simulation modeling and analysis* (1st ed.). New York, NY: McGraw-Hill Higher Education.

Law, A. M., & Kelton, W. D. (2000). *Simulation modeling and analysis* (3rd ed.). New York, NY: McGraw-Hill Higher Education.

Law, A. M., & McComas, M. G. (2001). Building valid models: How to build valid and credible simulation models. *Proceedings of the 2001 Winter Simulation Conference*, Arlington, VA, *1*, 22–29.

Lawrence, R. D., Hong, S. J., & Cherrier, J. (2003). Passenger-based predictive modeling of airline no-show rates. *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Washington, D.C.: *1*, 397–406.

Loerch, A. (2001). Introduction to computer simulation (class notes for OR540). Unpublished manuscript. Retrieved from http://classweb.gmu.edu/aloerch/simulation540.pdf

Lucas, T. W., & McGunnigle, J. E. (2003). When is model complexity too much? illustrating the benefits of simple models with Hughes' salvo equations. *Naval Research Logistics* (NRL), *50*(3), 197–217.

Metropolis, N. (1987). The beginning of the Monte Carlo method. *Los Alamos Science*, (Special Issue), 125-130.

Mitroff, I. I., & Featheringham, T. R. (1974). On systemic problem solving and the error of the third kind. *Behavioral Science*, *19*(6), 383–393.

Modeling and Simulation Coordination Office (MSCO). (2001). A practitioner's perspective on simulation validation. Unpublished manuscript. Retrieved from http://vva.msco.mil/Key/key-pr.pdf

Moore, G. (1975) Excerpts from a conversation with Gordon Moore: Moore's law. Intel Corporation.

Moore's Law. (2011). Moore's law. Retrieved from http://en.wikipedia.org/wiki/Moore%27s_Law

Nance, R. E. (1978). Model representation in discrete even simulation: Prospects for developing documentation standards. (Technical Report No. CS78001-R). Blacksburg, VA: Virginia Polytechnic Institute and State University.

Navy Modeling and Simulation Office (NMSO). (2009). Analytical models: Evolution and revolution. Retrieved from https://nmso.navy.mil/

Office of the Under Secretary of Defense (Acquisition, Technology and Logistics) (OUSD-ATL). (2006). Department of defense acquisition modeling and simulation master plan.

Padua, D. (2000, January/February) .The FORTRAN I complier. *Computing in Science and Engineering*, 70-75.

Parlar, M., & Sharafali, M. (2008). Dynamic allocation of airline check-in counters: A queueing optimization approach. *Management Science*, *54*(8), 1410–1424.

Schruben, L. (1983). Simulation modeling with event graphs. *Communications of the ACM*, *26*(11), 957–963.

Simio LLC (Simio). (2011). In Simio LLC (Ed.), *Simio reference guide*. Retrieved from http://www.simio.com/downloads/public/academic/SimioReferenceGuide.zip

Student. (1908). The probable error of a mean. *Biometrika*, *6*(1), 1–25.

United States Air Force. (2010, July). Air Force modeling and simulation vision for the 21st century. Headquarters, United States Air Force.

Wagner, H. (1969). *Principles of operations research with applications to management decisions*. Prentice-Hall.

White, K., Fu, M. C., & Sanchez, S. M. (2011). The winter simulation conference: The premier forum on simulation practice and theory. Unpublished manuscript. Retrieved from http://wintersim.org/ArticleWSC11.pdf

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Director, SEED Center
   Naval Postgraduate School
   Monterey, California

4. Dr. W. David Kelton
   Department of Operations and Business Analytics
   University of Cincinnati
   Cincinnati, Ohio

5. Dr. James R. Wilson
   Department of Industrial and Systems Engineering
   North Carolina State University
   Raleigh, North Carolina

6. LT Ben Anderson
   Naval Postgraduate School
   Monterey, California